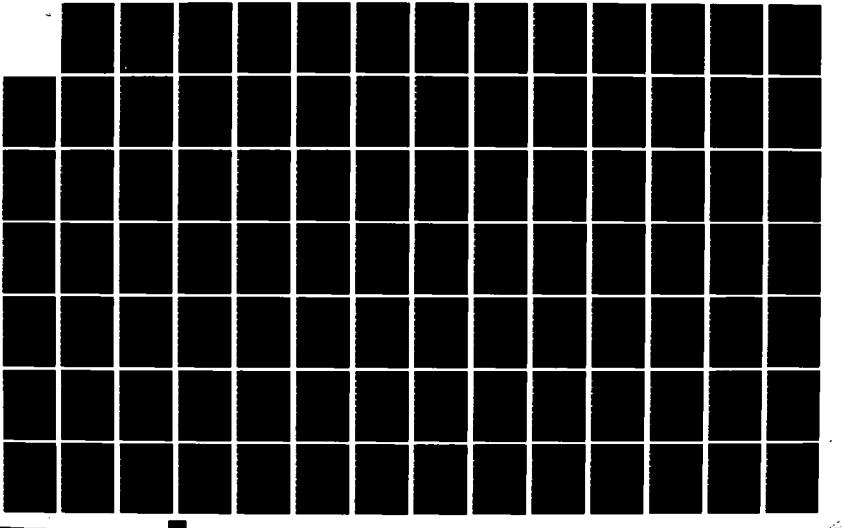
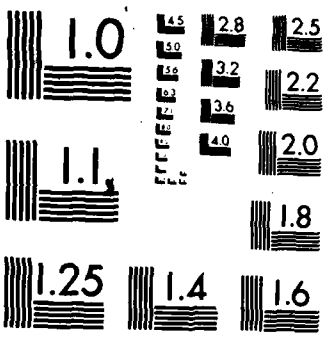
AD-A137 712 THERMAL VACUUM TEST FACILITY(U) NAVAL RESEARCH LAB WASHINGTON DC K M MILLER 31 JAN 84 NRL-MR-5217 1/2

UNCLASSIFIED F/G 14/2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NRL Memorandum Report 5217

Thermal Vacuum Test Facility

K. M. MILLER

Space Applications Branch
Aerospace Systems Division

January 31, 1984

NAVAL RESEARCH LABORATORY
Washington, D.C.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: NRL Memorandum Report 5217 — 2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): THERMAL VACUUM TEST FACILITY	5. TYPE OF REPORT & PERIOD COVERED: Final report
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): K. M. Miller	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Naval Research Laboratory, Washington, DC 20375	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 79-0733-03
11. CONTROLLING OFFICE NAME AND ADDRESS	12. REPORT DATE: January 31, 1984
	13. NUMBER OF PAGES: 111
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

NAVSTAR
GPS
Atomic clocks
Environmental testing

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An important part of development of electronic packages for space use is verification of performance under expected environmental conditions. The system developed at NRL is designed to provide thermal vacuum testing of atomic clocks for use in the Global Positioning System (GPS). It is capable of testing all pertinent clock parameters in a preprogrammed sequence. Test durations from minutes to months may be selected. Real time as well as historical data presentations and reductions are available.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CONTENTS

THERMAL VACUUM TEST FACILITY

1.0 Assembling the Equipment Rack

Equipment Needed:
==========

(does not include interface cards, cables, or software items)

1. 69 7/8 inch Cabinet Rack
2. Sliding Desk Top Shelf
3. Rack Mount Tray
4. Power Strips with at least 15 outlets
5. 9825 Calculator with flexible disk drive, plotter-general I/O-extended I/O, and string-advanced programming ROM's
6. 9878 I/O Expander
7. 9885M Flexible Disk Drive
8. 9885S Flexible Disk Drive
9. 98035 Real Time Clock
10. Racal-Dana Counter
11. Cs Control Box
12. 27V Power Supply
13. Chamber Temperature Controller
14. 436 RF Power Meter
15. 59301 ASCII to Parallel Converter
16. Analog Thermometer Matrix
17. 3455 Digital Voltmeter
18. 3495 Scanner
19. Kepco Bipolar Operational Power Supply
20. 7245 Plotter/Printer

[rack #2 only] 21. 1350 Graphics Translator

[rack #2 only] 22. 1311 Display

ASSEMBLY: This paragraph describes the assembly of the cabinet. Two different versions of the rack can be built. The 1st rack contains items 1-20 from the equipment list above. These items are arranged in the cabinet as shown in figure 1.1. Item 20, the plotter/printer is not rack-mountable, but can be placed on a table top. The 2nd rack contains the items marked [rack #2 only], in addition to items 1-20. The graphics translator and display generate soft-copy lists and plots and are driven by the same software that runs the 1st rack. The equipment does not all fit into rack #2. See figure 1.2. The graphics translator is placed in slot reserved for the calculator in rack #1. The calculator, display, and plotter/printer are placed on a table top.

SELECT CODES and DEVICE ADDRESSES: Each device has its own unique select code and device address. The select code is a single digit from 0 to 9 specified on the bus interface card by adjusting a rotary select switch. The device address is a two digit number from 0 to 31 specified by selecting the appropriate bits on a dip

Manuscript approved September 22, 1983.

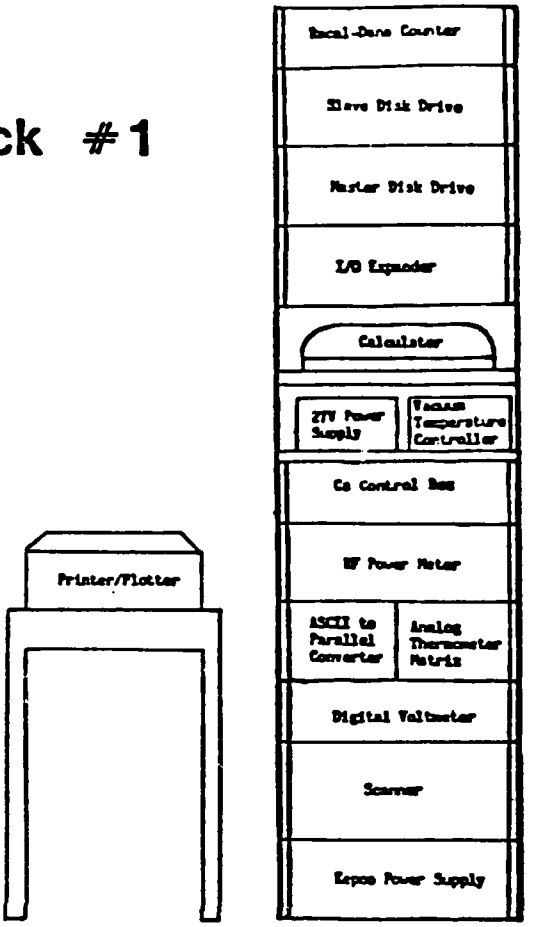
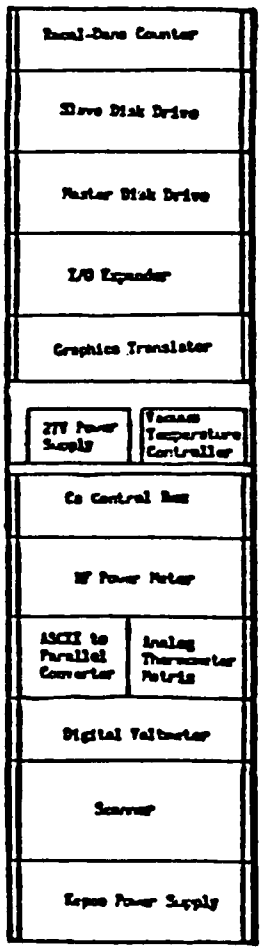
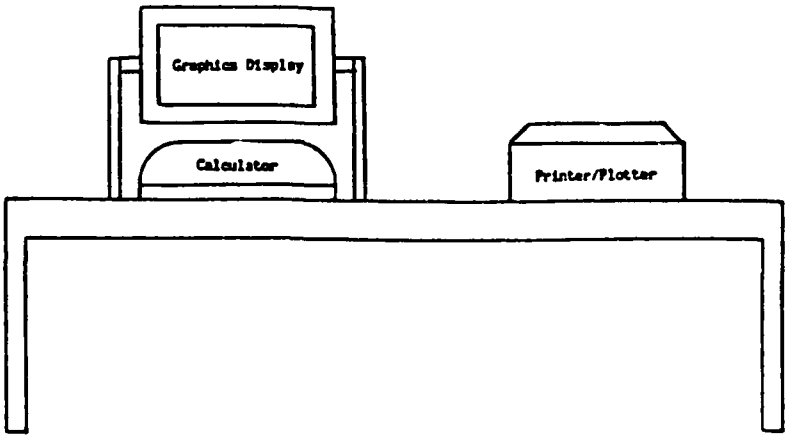


Figure 1.1



rack #2

Figure 1.2



switch. The dip switch is usually located on the back panel of an instrument, but sometimes is placed on a board and is accessible only by taking the cover off. Three different buses are used in this system. Each bus requires its own slot in the I/O expander box. The 98035 HP-IB interface is set to SC 7, the 98032 opt 085 disk interface is set to SC 8, and the 98035 real time clock, which requires its own slot, is set to SC 9. All the instruments which communicate with the calculator, except the clock and disk drives, talk over the HP-IB. The device addresses for each instrument are listed in table I. Although the RF power meter is a bus-compatible instrument, it is not placed on the bus. Rather, the recorder output on the back panel is sampled by the voltmeter. The master disk drive is set to drive 0 by a rotary select switch on the back panel of the instrument. The slave disk drive is set to drive 1.

Table I: Select Codes and Device Addresses

SC/DA	Bit Pattern	Access	Device	Drive
701	100001	inside	Kepco Power Supply	-
705	00101	back panel	7245 Plotter/Printer, plot mode	-
706	00101	back panel	7245 Plotter/Printer, print mode	-
707	0000111	back panel	59301 ASCII to Parallel Converter	-
709	01001	inside	3495 Scanner	-
712	001100	back panel	Racal-Dana Counter	-
718	10010	inside	1350 Graphics Translator	-
722	0010110	back panel	3455 Digital Voltmeter	-
8	-	-	9885M Flexible Disk Drive, master	0
8	-	-	9885S Flexible Disc Drive, slave	1
9	-	-	98035 Real Time Clock	-

2.0 Wiring the Interconnections

AC POWER: Connect AC power cords from the back of each instrument to the power strip. Connect the power strips together so that only one line runs out to the wall outlet. Plug the 27V power supply into the duplex power outlet on the front panel of the vacuum chamber.

BUS CONNECTIONS: Plug the I/O expander card into one of the three calculator slots. Cover the other two slots with plastic inserts, if possible. Plug the 98032, 98034, and 98035 cards into any of the I/O expander slots. Connect the other end of the 98032 to the master disc drive. Lock it into place by sliding the clip to the right. Connect the master disk to the slave disk by attaching the special mating cable. Both connectors on the slave disk are identical, so either one can be used. This cable also locks into place at each end. Connect the other end of the 98034 to the counter. Now attach the other instruments listed in table I, with a select code of 7, to the bus with short HP-IB cables. Be aware of creating bus loops. Do not connect the RF power meter to the bus.

INSIDE THE CHAMBER: Five lines must be connected inside the chamber. The 1st line runs from a 37-pin Deutsch connector attached to the chamber bulkhead, to the analog thermometer matrix receptacle box mounted on the chamber base plate. The 2nd cable has a 37-pin Duetsch bulkhead connector on one end, and divides into two separate lines at the other end, which are each terminated by Bendix connectors. These terminals are attached to their mates on the Cs clock. The 3rd line is a coaxial cable. It runs from a bulkhead BNC connector on one end, to the SMA RF output port on the Cs clock at the other end. The 4th and 5th leads supply power to the thermoelectric device (TED). Select wire sufficient to carry 5A without heating. These two wires each have BNC terminations at one end for connecting to the bulkhead feed-throughs, and are attached at the other end to a 9-pin amphenol female connector, pins 2 and 4. The amphenol connector mates with a male connector wired to the TED.

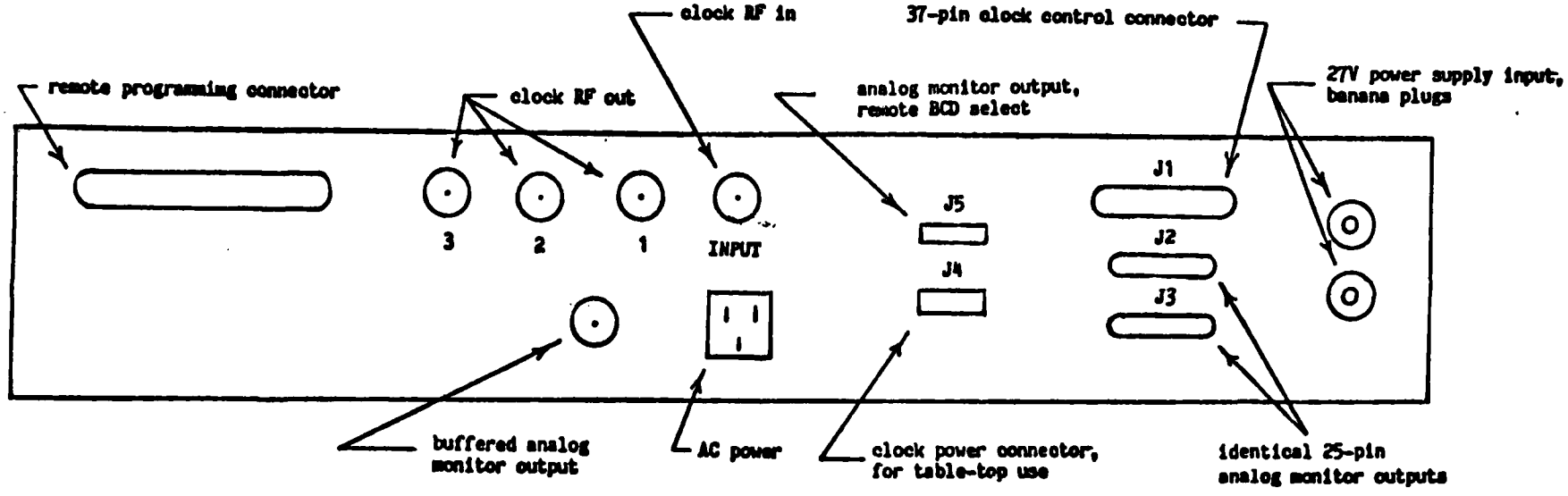
OUTSIDE THE CHAMBER: The 5 lines from above are brought through the bulkhead and connected to appropriate devices. The 1st cable is about 15' long, has a 37-pin Deutsch connector at one end and 10 3-conductor phone plugs at the other end. Attach the Deutsch connector to the 37-pin port which feeds through to the receptacle box. The phone plugs at the other end of the cable are numbered 0 through 9. Each of them mates with a phone jack on the back of the thermometer box. The 2nd cable is also about 15' long, has a 37-pin Deutsch connector at one end and a 37-pin amphenol male connector at the other end. Attach the Deutsch connector to the bulkhead feed-through port that mates with the clock cable on the inside. Attach the other end to the 37-pin connector on the back of the Cs control box. The 3rd cable, a segment of coax, runs

from the RF feed-through BNC on the chamber bulkhead, to the input BNC on the back of the Cs control box. The 4th and 5th lines have to be the same gauge as used inside. Run them from the appropriate BNC feed-throughs to the terminal strip pins on the back of the Kepco power supply labeled OUT and COM.

COUNTER CONNECTIONS and SWITCH SETTINGS: The counter compares the clock under test with the in-house reference oscillator. It does this by measuring the time interval between comparable points on the two waveforms. By analyzing how time interval changes as a function of time, frequency stability can be determined. First connect a coaxial cable from one of the three Cs control box BNC output ports to the BNC connector on the back of the counter labeled 'A'. Then run a cable from room 204 containing the 10.23 MHz reference into the BNC connector labeled 'B'. Set the gate select switch to NORM. All other connectors should be left blank, unless an external reference is used.

Cs CONTROL BOX CONNECTIONS: Not all of the connectors on the back of the Cs control box are used. See figure 2.1 for connector locations. To begin, turn on the 27V power supply and adjust the output to 27V. Turn it off, and connect the minus side to ground. Connect the power supply terminals to the Cs control box with 2 banana plug wires, (+) to red and (-) to black. Connect the 15' Cs clock cable to J1. Connect the special scanner cable, which has a 25-pin amphenol connector at one end and 25 lines terminated in amp lugs at the other end, to J2. See SCANNER CONNECTIONS for instructions in wiring that cable to the scanner. Connect J3 to the MASDAT system in room 204 using the special cables that have been run for that purpose. Leave J4 and J5 blank. Feed the RF output of the clock into the BNC labeled 'INPUT' with a coaxial cable that runs from the appropriate feed-through on the chamber bulkhead. The BNC's labeled '1', '2', and '3' are the output terminals of a power splitter. Send one up to channel A on the back of the counter, one to the MASDAT system in room 204, and one to the power sensor attached to the RF power meter. The buffered analog output and remote programming connectors are not used.

CHAMBER TEMPERATURE CONTROLLER CONNECTIONS: The temperature controller works by producing a floating voltage equivalent to the voltage a type J thermocouple would develop at the setpoint temperature. This voltage is compared in the Honeywell controller with a voltage generated by the base plate thermocouple. The difference is used to select heating or cooling mode. Connect a short length of cable terminated by two 50-pin male HP connectors from the back of the vacuum chamber temperature controller to the J1 connector on the back of the ASCII to Parallel converter. Attach a cable from the phone jack to terminals 13 and 14 on the back of the Honeywell controller. The center lead of the phone plug, corresponding to the tip, connects to terminal 14. The middle of the phone plug connects to terminal 13. The phone plug ground lead, the section in contact with the chasis, is not used.



cesium control box, back panel

Figure 2.1

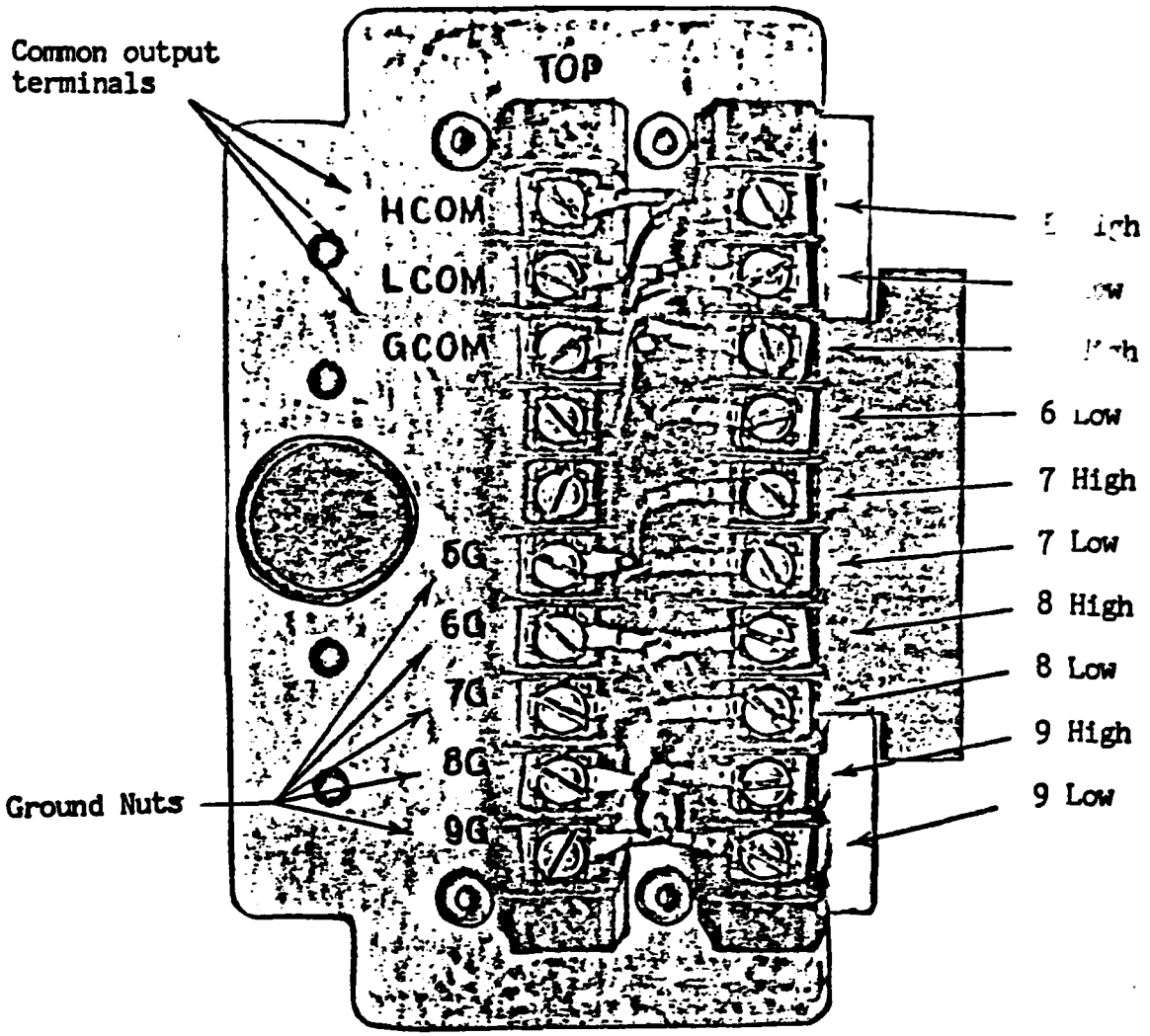
RF POWER METER CONNECTIONS AND SETTINGS: An HP 8481 Power Sensor probe must be connected to the power meter. If the power has a rear terminal sensor input for this probe, use it. Otherwise, connect the probe to the front panel. The end of the sensor should be attached to one of the three RF output ports on the Cs control box. Direct the recorder output on the back of the power meter to scanner channel #26. The HP bus and RF blanking connections are not used. Set the front panel controls to dBm mode and 100% cal function.

ANALOG THERMOMETER MATRIX CONNECTIONS: The analog thermometer matrix accepts a BCD digit from the ASCII to Parallel converter, relays the leads of one of ten input thermistors into a thermometer, and sends the output voltage into one of ten lines for recording by the voltmeter. Connect the plugs numbered 0 through 9 on the thermometer cable to the proper phone jacks on the back of the thermometer box. Next, connect the 9-pin amphenol connector on the thermometer to connector J2 on the ASCII to Parallel converter. A special cable has been made for this purpose. Finally, connect the 15-pin amphenol connector on the thermometer to the 4th scanner decade with another special cable made for this purpose. The end of this cable contains 11 wires terminated in amp lugs. Pin #1 on the thermometer connects to scanner channel #30, pin #2 to 31 ..., pin #10 to 39. Pin #11 connects to the common low on the scanner decade. The 4th scanner decade is dedicated to temperature measurements.

ASCII TO PARALLEL CONVERTER CONNECTIONS: The ASCII to Parallel box is dedicated to controlling both the vacuum chamber temperature and the analog thermometer matrix. The 1st 10 digits of any word sent to the ASCII box exit connector J1. The temperature controller uses 2 1/2 of these digits plus a sign bit. The two boxes are connected together by a 50-wire cable terminated in male HP connectors. The 2nd 10 digits sent to the ASCII box exit connector J2. The analog thermometer matrix uses 1 digit. The ASCII box and thermometer are connected together by a special cable that has a 50-pin male HP connector at one end, and a 9-pin female amphenol connector at the other end.

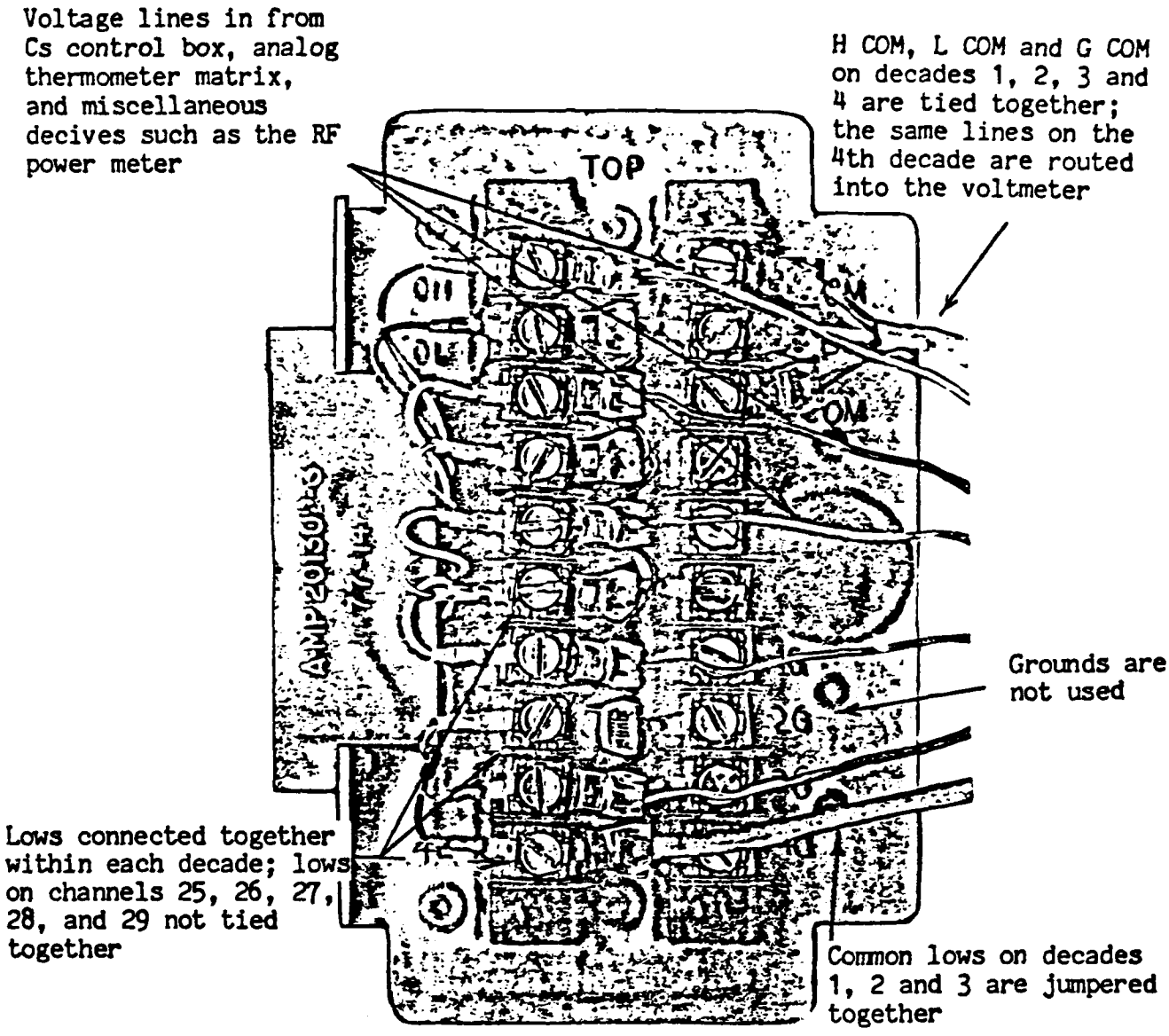
VOLTMETER CONNECTIONS and SWITCH SETTINGS: There are 5 banana plug connectors on the back of the voltmeter. The 2 on the left, under the 4 wire ohm label, are not used. HI connects to H COM on scanner decade #4. LOW connects to L COM, and GUARD connects to G COM. Special 3-wire cables with banana plugs at one end and amp lugs at the other end have been constructed for this purpose. Set the rear panel input select switch to REAR.

SCANNER CONNECTIONS: The scanner accepts 40 analog voltage signals on its four 10-terminal connector cards; and channels these signals into the digital voltmeter under calculator control. An example of an unwired scanner terminal card is shown in figure 2.2. Each scanner card has 10 High, 10 Low and 10 Ground connections. In addition each card has two Common High, Common Low



scanner card without connections

Figure 2.2



scanner card with connections

Figure 2.3

and Common Ground output connections. The terminals are numbered 0 through 9. The card which plugs into the scanner slot farthest to the right has a ten's digit of 0. The next one over is 1, the one after that 2, and the last card 3. Therefore, the scanner channels are numbered 0 through 39. When the calculator selects channel 12, for example, the 3rd set of relays on the 2nd decade is closed.

The 1st 25 relays, numbered 0-24, are reserved for the Cs clock monitors. A special cable has been constructed with a 25-pin female amphenol connector at one end, and 25 wires terminated in amp lugs at the other end. The amphenol connector plugs into connector J2 on the Cs control box. The 25 amp lugs are wired to the High side of the 1st 25 channels on the 1st 2 1/2 scanner cards. Pin #1 connects to channel 0, pin #2 to channel 1, ... pin #11 to channel 10, ... pin #25 to channel 24. The Low sides of the 1st 25 channels are all tied together. To establish continuity between the Low coming out of the Cs control box and the Low on the scanner cards, place a jumper between channel 12 High and channel 12 Low. Pin #13 coming out of the Cs control box is analog ground.

The last 10 relays, numbered 30-39, are reserved for temperature measurements. Channel 30 measures the temperature of the thermistor plugged into jack 0 of the thermometer receptacle box located inside the chamber. Channel 31 measures the temperature on thermistor #1, ... channel 39 measures the temperature on thermistor #9. A special 11-wire cable has been made for the purpose, with a 15-pin female amphenol connector at one end and amp lugs at the other end. Pin #1 connects to channel 30, pin #2 connects to channel 31, ... pin #10 to channel 39. All the Lows on the 4th decade should be jumpered together. Do not connect the Lows on channels 30-39 to the Lows on channels 0-24.

The remaining 5 channels, 25-29, are available for additional recordings. The following devices will generally be connected: temperature offset from pins 16 (low) and 17 (high) on the back of the Honeywell controller into channel 25; the RF power meter recorder output into channel 26; and the ionization gauge recorder output, if available, into channel 27.

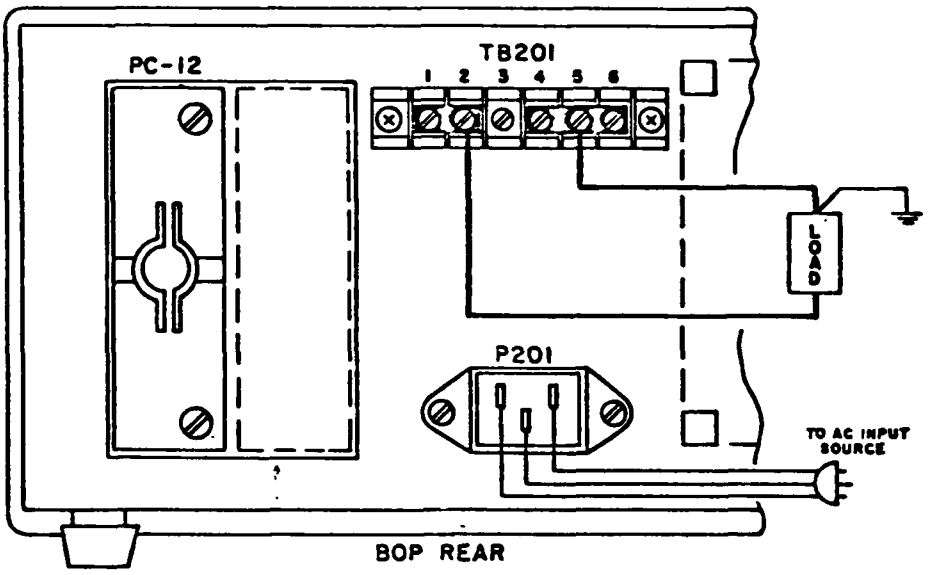
H COM, L COM and G COM on each of the 4 decades should be connected together as shown in figure 2.3. The same terminals on the 4th decade must be connected to the HI, LOW and GUARD inputs on the back of the voltmeter. A special 3-wire cable has been made for this purpose, with amp lug terminations at one end and banana plug terminations at the other end. The names and functions of each of the channels are assigned during a software initialization routine.

KEPCO POWER SUPPLY CONNECTIONS: The Kepco power supply is a bipolar 36V @ 12A, HP-IB programmable device. It is used to drive current through the thermoelectric device. In order to use the rear terminal strip, the front panel error sensing links must be removed. Short rear terminals 1 and 2 together; and 4, 5 and 6

together, as shown in figure 2.4. Connect the TED leads to terminals 2 and 5.

PLOTTER/PRINTER SETTINGS: Set the dip switch on the rear of the plotter/printer to: ERROR BEEP, 8 LPI, 8 BIT, METRIC and 9872.

CONNECTIONS TO ROOM 204: The clock RF output and analog voltages go out to the MASDAT system in room 204. A 10.23 MHz clock reference signal comes in. The RF output signal comes out of a BNC on the back of the Cs control box. The analog voltages come from connector J3 on the Cs control box. The 10.23 MHz clock reference signal is applied to channel 3 on the counter.



NOTE: FRONT SENSING LINKS MUST BE REMOVED. REAR SENSING LINKS MUST BE CONNECTED AS SHOWN.

bipolar operational power supply

Figure 2.4

3.0 Instrument Startup and Checkout

Before starting a test, the following steps should be taken to insure that each device is calibrated, receiving the appropriate signals, and operating properly.

MOUNTING THE CLOCK: Bolt the bottom of the clock to the upper plate of the TED. Bolt the lower plate of the TED to the base plate of the chamber. Connect the 9-pin male amphenol connector on the TED to its mating connector inside the chamber. Attach the RF output, power, and analog voltage monitor lines to the 3 ports on the front of the clock.

27V POWER SUPPLY: The AC power cord should be plugged into the duplex power outlet on the front panel of the chamber. If the foreline pressure is low enough, duplex power will come on when the push-button next to the outlet is pressed. If duplex power will not come on, plug the AC power cord temporarily into a wall outlet. Disconnect the banana plugs from the power supply and turn it on. Adjust the output to 27V. Turn the power supply off. Reconnect the banana plugs and return the AC power cord to the duplex power outlet. Set the red trip-point needle on the foreline pressure gauge to 100 millitorr.

Cs CONTROL BOX: Check for proper connections from the back of the Cs control box to the clock, counter, RF power meter, and MASCAT system. Turn the 27V power supply on. Refer to the manufacturer-supplied documentation for starting the clock. The procedure varies from manufacturer to manufacturer, and from model to model. Check the voltage monitors to see whether they are within manufacturer-specified limits. Before roughing the vacuum system, be sure to disconnect power from the clock.

COUNTER CHECKOUT: With the clock powered up, check the frequency on channel A by setting the appropriate counter keyboard switches to NORM and SEP, and pressing the key labeled 'FA'. If the display does not show the correct clock frequency, something is wrong. Check the signal coming over from the in-house reference standard in room 204 by temporarily switching that cable from channel B into channel A. Display the frequency by pressing FA. The two signals should have the same frequency. Return the clock signal to channel A and the reference signal to channel B.

REAL TIME CLOCK CHECKOUT: To load the correct time into the Real Time Clock enter:

```
wrt 9, "SMM,DD,HH,MM,SS"      <execute>
```

where MM,DD ... SS are the month, day, hour, minute and second. To read the clock enter:

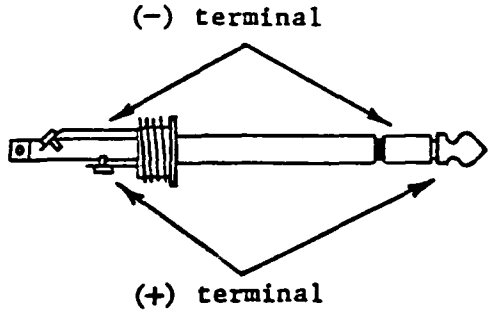
```
dim T$[14]                     <execute>
```

```
wrt 9, "R"                    <execute>
red 9, T$                     <execute>
dsp T$                        <execute>
```

If the calculator hangs up while reading the clock, check the connection of the clock to the I/O expander box. Wait a few minutes and try again. If the problem persists, replace the clock card with a new one.

CHAMBER TEMPERATURE CONTROLLER and HONEYWELL CONTROLLER, CALIBRATION and CHECKOUT: Before doing anything in this section, turn the chamber Thermal Control off (push-button on the front panel of the chamber). To calibrate the chamber temperature controller, remove the top cover and locate the trim pot. Disconnect the plone plug from the phone jack on the rear of the box, and put an empty phone plug, with plastic cover removed, back in its place. Connect a coaxial cable with aligator clip leads from the digital voltmeter to the two terminals of the phone plug as shown in figure 3.1. Set the voltmeter for front panel operation.

Figure 3.1:



phone plug

Turn the temperature controller on. The largest number the D/A converter in the controller can accept is 1111111111 binary, or 3FF hexadecimal. On the calculator enter:

```
wrt 707, "+000"               <execute>
```

This should produce an output of 0.000 mV. Next enter:

```
wrt 707, "+3FF"               <execute>
```

Adjust the trim pot until the output is 8.000 mV. As a quick check that everything is working, enter:

```
wrt 707, "-3FF"               <execute>
```

Now the output should be -8.000 mV. Reconnect the phone plug and replace the cover.

Loosen the case lock screw on the Honeywell controller and slide the box out far enough to reach the LOCAL/REMOTE switch. Place the Honeywell in LOCAL mode. With the chamber Thermal Control still off, turn the dial clockwise and counterclockwise, and watch the red alarm light come on and off. The red light comes on whenever the chamber temperature is outside a narrow setpoint temperature range. Adjust the HI and LOW alarm screws on the side of the Honeywell until the red light comes on at the +2 and -2 percent deviation points (one tic mark off center on the controller dial). Place the honeywell controller in remote. Push the case back in and tighten the case lock screw. On the calculator enter:

wrt 707, "+3FF" <execute>

The red temperature offset indicator needle on the Honeywell controller should peg to the left. Then enter:

wrt 707, "-3FF" <execute>

which should make the needle peg to the right. If this does not happen, check the connection from the temperature controller to the Honeywell controller. Before a test is run, turn the chamber Thermal Control on.

SCANNER and VOLTMETER CHECKOUT: Before starting this section, check the wiring and set the rear panel input selector switch on the voltmeter to REAR.

1) Cs CONTROL BOX signals - turn the clock on. On the calculator enter:

   wrt 709, "C00E" <execute>

   This command opens all scanner channels and closes channel 0. Verify that channel 0 displays a value appropriate for Supply V. Now enter:

   wrt 709, "C01E" <execute>

   to close channel 1. Verify that channel 1 displays a value appropriate for Supply I. Continue this procedure for all 10-12 channels in use. It may be helpful to obtain a printer listing of the INIT file as an aid to the identification of each of the channels.

2) HONEYWELL CONTROLLER signal - the deviation recorder output of the Honeywell controller is connected to channel 25. The output signal is a +/- 1V dc signal proportional to deviation from setpoint. Enter:

   wrt 709, "C25E" <execute>

   and check the output by turning the Honeywell dial to

deflect the error needle.

3) RF POWER METER signal - the recorder output on the power meter is connected to channel 26 of the scanner. The output signal is +1.000V dc full scale for the range selected. With the clock on, enter:

```
wrt 709, "C26E"                    <execute>
```

and verify that the output power is in the appropriate range.

4) IONIZATION GAUGE signal - if the ionization gauge on the vacuum chamber control panel has a recorder output, it should be connected to channel 27. Output is +2V full scale. Logarithmic changes in pressure produce linear changes in voltage. To test the output of the gauge, the chamber must be pumped down, and the ion tube turned on. Enter:

```
wrt 709, "C27E"                    <execute>
```

and check the output voltage.

5) ANALOG THERMOMETER MATRIX signals - Before proceeding with this section, turn the Thermal Control, on the chamber control panel, off. The analog thermometer expects to receive a non-complemented BCD character from 0 to 9. The ASCII to Parallel box is output-formatted to generate complementary hexadecimal. To write a 0 to the thermometer, an 'F' must be sent to the ASCII box in column 11. To write a 1, an 'E' must be sent, and so on. To write a 9, the character '6' must be sent. The transforming function is: hex(15 - the channel # desired). If a thermometer jack is left empty, and that channel is selected, an output voltage of -.51 mV will be displayed. Otherwise, the displayed voltage should equal the temperature at the thermistor divided by 100. To check thermistor 0, enter on the calculator:

```
wrt 707, "F0000000000"             <execute>
wrt 709, "C30E"                    <execute>
```

The first line selects thermistor 0, and the second line channels the output voltage of the thermometer into the voltmeter. To check thermistor 1, enter:

```
wrt 707, "E0000000000"             <execute>
wrt 709, "C31E"                    <execute>
```

and so on, decrementing the 1st character in the wrt 707 statement, and incrementing the 3rd digit in the wrt 709 statement, with each increment in channel number. To

check the last thermistor, enter:

```
wrt 707, "60000000000"        <execute>
wrt 709, "C395"               <execute>
```

With each channel selected, check the appropriateness of the number displayed on the voltmeter.

KEPCO BIPOLAR POWER SUPPLY and THERMOELECTRIC DEVICE CHECKOUT: Words written to the Kepco power supply are 6 characters long, formatted as follows: CMMMLL, where C is a mode command from 0 to 7, MMM is the output select, and LL is a limit. The system uses the Kepco in High Current mode which provides +/- 12 A full scale. The command for positive polarity is C=4, and for negative polarity, C=5. MMM ranges from 000 to 999. 000 corresponds to 0 Amps output, 999 corresponds to 12 Amps output. Each increment of 083 to MMM adds 1 A of output current. In current mode the device is voltage limited. The system prevents the current from going above 4 Amps. But as a safety precaution, the device is voltage limited to a value corresponding to about 5 Amps. LL ranges from 0, which corresponds to 0 V, to 99, which corresponds to 36 V. LL is always set to 60 (about 22 V).

Check the wiring to the TED and enter:

```
wrt 701, "416660"             <execute>
```

The output should indicate +2 Amps. With this polarity, the upper plate of the TED should heat. If, instead, it cools, turn the Kepco off and interchange the lead wires attached to the outside ports on the chamber bulkhead. Turn the Kepco back on and repeat the process. To cool the TED, enter:

```
wrt 701, "516660"             <execute>
```

Never send more than 4 Amps current to the TED (MMM=333).

4.0 Hardware Configuration

The Thermal Vacuum Test Facility is a calculator-controlled, automatic data collection system capable of controlling the thermal environment within a vacuum chamber. The calculator communicates directly with a peripheral I/O expander as shown in figure 4.1. The I/O expander controls 3 buses, each with a unique select code. The general purpose interface bus (GPIB), to which most of the instruments are connected, is assigned a select code of 7. The two disk drives are on a separate bus with a select code of 8. The real time clock occupies another bus slot, with a select code of 9. Instruments on the GPIB are assigned a unique device address from 0-31. The system operates according to a multiple interrupt scheme. Data collection has the highest priority. TED control has the next highest priority. On-line data reduction programs have the lowest priority.

Temperature is controlled, both at the level of the chamber base plate, and at the level of the TED. The chamber temperature controller programs the Honeywell controller to set the base plate temperature to whatever value is desired. Oscillations in base plate range from 3-10 degrees C, depending on the setpoint temperature chosen. The Kepco power supply fine-tunes the temperature to within 3 tenths of a degree C, eliminating the effect of base plate error. The TED can support a temperature gradient of 15 degrees C.

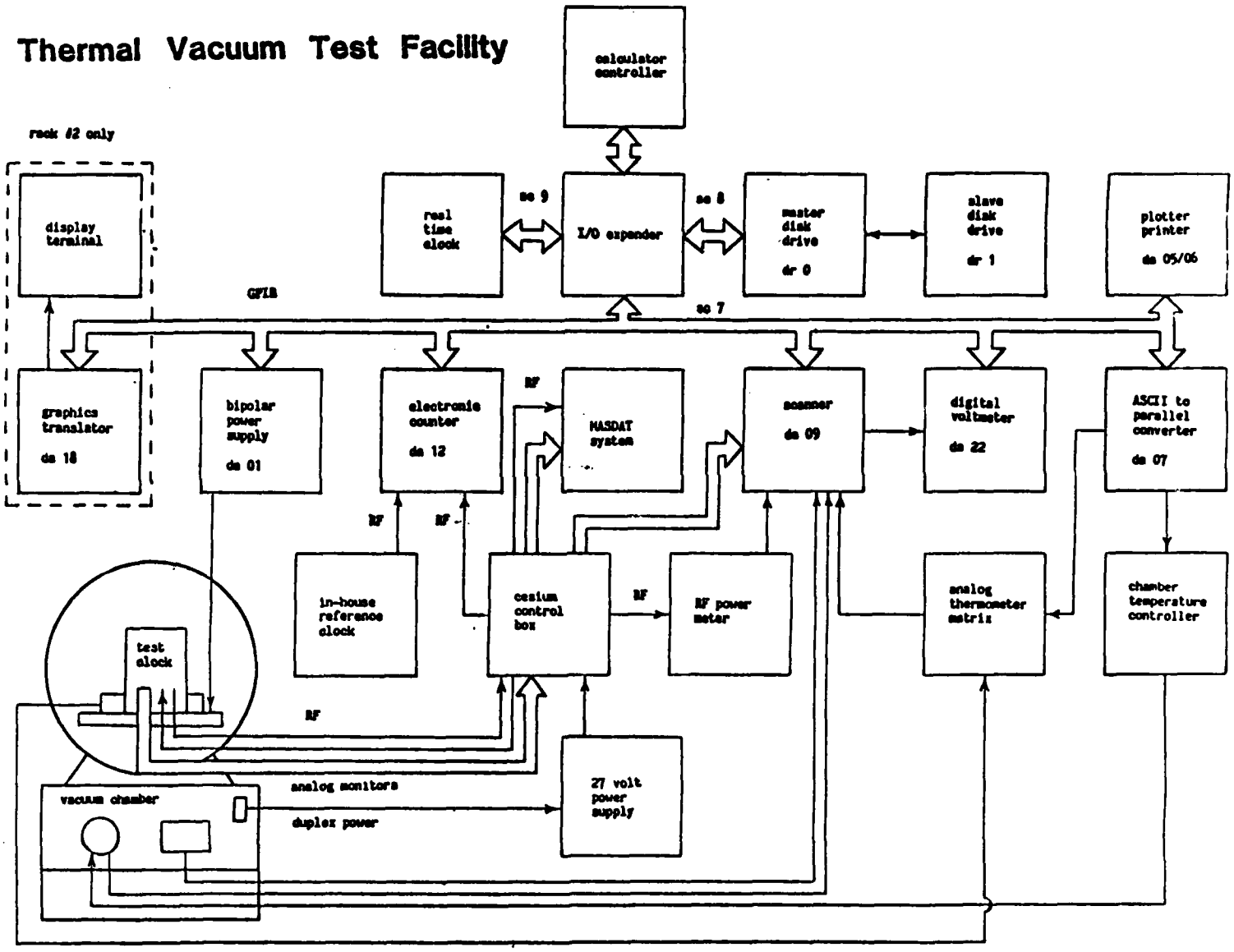

Thermal Vacuum Test Facility
calculator controller
rack #2 only
display terminal
real time clock
sc 9
I/O expander
sc 8
master disk drive
dr 0
slave disk drive
dr 1
plotter printer
da 05/06
GPIB
sc 7
graphics translator
da 18
bipolar power supply
da 01
electronic counter
da 12
RF
HASDAT system
scanner
da 09
digital voltmeter
da 22
ASCII to parallel converter
da 07
in-house reference clock
cesium control box
RF power meter
analog thermometer matrix
chamber temperature controller
test block
vacuum chamber
analog monitors
duplex power
27 volt power supply

Figure 4.1

5.0 Software Configuration

There are two programs a user should be familiar with. The first is INIT, a program that edits initialization files containing temperature control, scanner control, and measurement interval data. INIT is stored on the system disk, loaded in drive 0. To run INIT, enter:

```
drive 0                    <execute>
get "INIT"                 <execute>
                           <run>
```

A flow diagram showing the questions prompted by INIT is shown in figure 5.1. INIT, like many programs in the system, requests a binary cassette tape labeled the 'Softcopy Graphics Library'. This tape contains programs for facilitating output to the graphics translator. Insert the tape into the cassette drive and press <cont>.

Two data files are maintained by the system. They are: INIT.D and INIT.V. The 1st is a default initialization file containing standard settings as shown in figure 5.2. INIT.D can be edited, if necessary, but it would be wise not to change it. The other file, INIT.V, is a variable initialization file for tests which require non-default conditions. INIT.V should be updated each time the requirements of a test change.

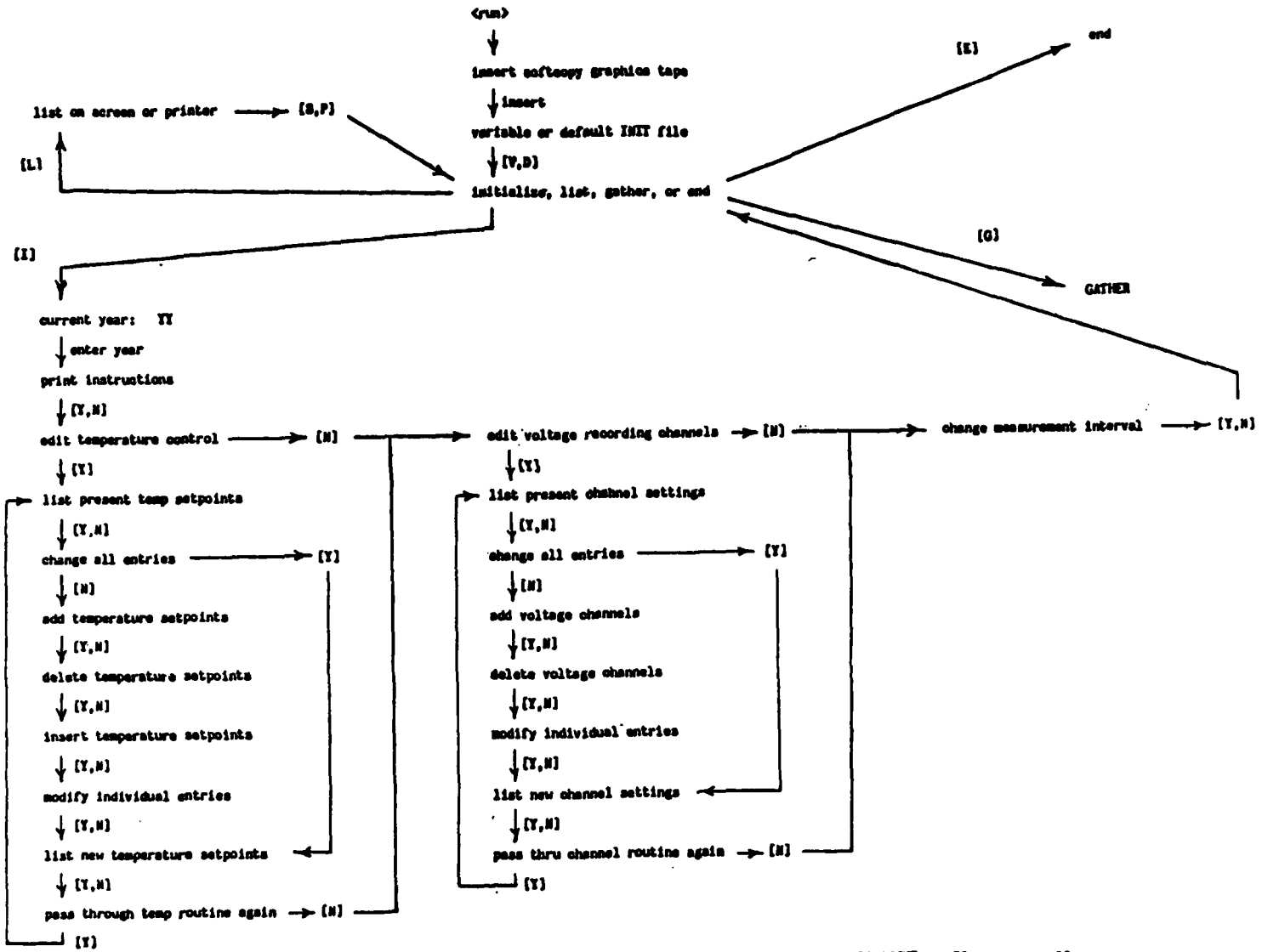
To edit one of the initialization files, enter 'I' to the prompt, 'Initialize, List, Gather, or End'. Flow is directed through 3 program segments as shown in figure 5.1: temperature control, scanner control, and measurement interval. Prompts should be answered with YES or NO unless otherwise stated. Entering 'Y', for example, to the prompt 'delete temperature setpoints' will enter the user into a subroutine that allows for selective deletion of temperature setpoints.

To list the contents of an initialization file, enter 'L' to the prompt, 'Initialize, List, Gather, or End'. The program will prompt for screen or printer. Enter 'S' for screen only if the graphics translator and display are on the system [rack #2].

Entering 'G' to 'Initialize, List, Gather, or End' will call the program GATHER into calculator memory, and start its execution. Entering 'E' will end INIT.

GATHER is the system's data collection program. It resides in calculator memory whenever a test is running. GATHER can be requested in INIT, or it can be run by entering:

```
drive 0                    <execute>
get "GATHER"               <execute>
```



INIT flow diagram

Figure 5.1

Thermal Vacuum Test Facility

Program: INIT
Written: 12-17-80
For: Space Applications Branch
US Naval Research Lab

Revision Date: 07-28-83
Measurement Interval: 0d 1h 0m 0s
Counter Gate Time: 100 msec

****** Temperature Control ******

##	Temperature (deg C)	Duration Days	Hrs	Mins	Secs
1	25	30	0	0	0

****** Scanner Control ******

##	Channel Number	Channel Name	Channel Function
1	0	Supply I	DC V
2	1	Supply V	DC V
3	2	Ionizer	DC V
4	3	Freq Mult	DC V
5	4	Qtz Oven	DC V
6	5	Ion Pump	DC V
7	6	Cs Oven	DC V
8	7	Beam I	DC V
9	8	C Field	DC V
10	9	Control	DC V
11	10	EM	DC V
12	12	Ground	DC V
13	25	Tmp Offset	DC V
14	26	RF Power	DC V
15	27	Vacuum	DC V
16	30	TED Temp	DC V
17	31	BP Temp	DC V

INIT.D default initialization file

Figure 5.2

<run>

A flow diagram for GATHER is shown in figure 5.3. GATHER requests the Softcopy Graphics Library tape, asks for the initialization file to use. When the calculator prompts for a test start date, entering <cont> will cause the current date and time to be loaded. An end date must be specified.

The POWER FAIL TAPE is a special cassette tape that remains in the cassette drive whenever a test is running. The tape contains two files. The 1st is a program file containing the program REBOOT. The second is a short data file. Whenever power is turned on, the calculator attempts to load and run a program from file 0 of the cassette tape. When REBOOT is loaded after a power failure, it resets all of the devices in the system, restarts the interrupt cycles, and loads in the memory file MEMORY. MEMORY contains an exact image of calculator memory at the time it was saved. The memory file is resaved each time data is collected. To create a POWER FAIL TAPE, load a blank cassette tape into the cassette drive and enter:

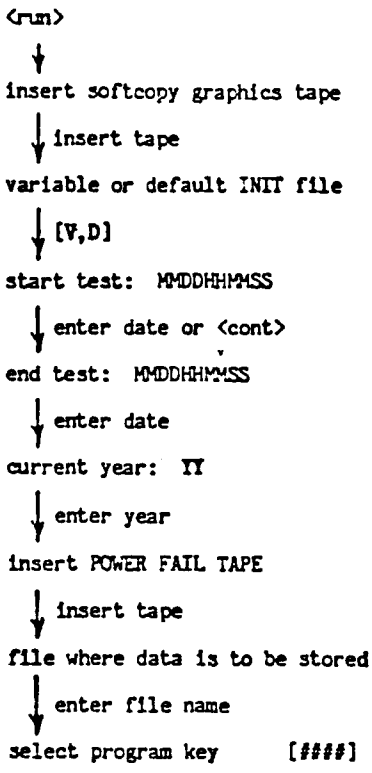
```
ert 0                          <execute>
rew                            <execute>
mrk 1, 2000                    <execute>
mrk 1, 100                     <execute>
drive 0                        <execute>
get "REBOOT"                   <execute>
rcf 0                          <execute>
```

Be sure to label the tape, 'POWER FAIL'.

After loading the POWER FAIL TAPE, the calculator requests a data file name. File names are 6 characters or less. A very common error is to attempt opening a file for which there is not enough disk space. If this operation produces 'error 08' (insufficient storage space on disk), correct it by killing unnecessary data files, going to a new data disk, or shortening the time interval of the test so that not as much data is collected. Once GATHER is successfully running it displays, 'select special function key [####]' where #### is the number of measurement cycles remaining.

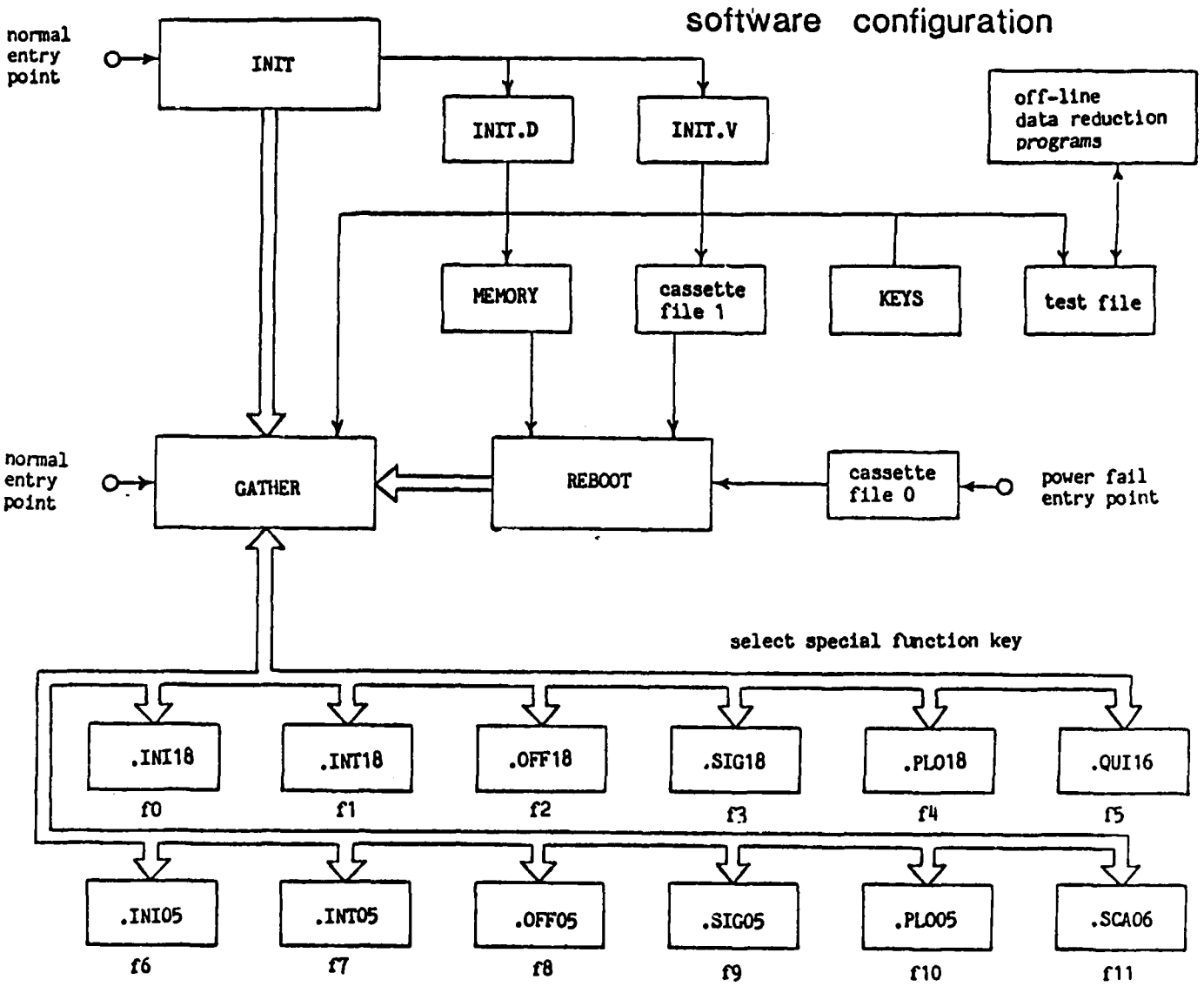
Figure 5.4 shows how the system programs interact. The small boxes denote data files and subroutines. Information flow is indicated by the arrows. An arrow to a file, such as INIT.V, indicates that a program writes to that file. An arrow away from a file, such as KEYS, indicates that the file is read from. There are three ways of entering the system: 1) through INIT, 2) through GATHER, and 3) through a power fail.

GATHER controls twelve subroutines. Subroutines are swapped in and out of calculator memory at user request. The subroutines perform data handling routines while data is still being collected.



GATHER flow diagram

Figure 5.3



Thermal Vacuum Test Facility

Figure 5.4

By convention, the names of the subroutines on the disk all begin with a decimal point. They do not run in stand-alone mode because they lack variable and device declaration statements. They operate only when GATHER is running. To execute a subroutine during a test, press the special function key associated with that function. Some subroutines require additional user information. This is also entered by pressing special function keys. See the section on "Obtaining data during a test" for more information regarding subroutines.

GATHER operates two programmed interrupts. The 1st is a variable duration interrupt which controls the interval between measurement cycles. The 2nd interrupt is fixed at 5 seconds, and controls the TED. It has a lower priority than the 1st. It is possible to observe the effect of the interrupts by observing the digital voltmeter display.

All the remaining programs in the system operate independently of GATHER. They may be loaded into the calculator when a test is through. Section 9 contains program descriptions and sample outputs.

All programs that output data to the plotter/printer or graphics display terminal have names that are formatted in the following manner: .NNNDD, where the decimal point is present only if the file is a subroutine, NNN is is a 3-character mneumonic, and DD is the device address of the output terminal.

6.0 Running a Test

To run a clock test, follow the sequence of steps listed below:

1) Verify that all the appropriate devices are in the system, powered on, and working correctly (refer to sections 1 and 3).
2) Verify that the proper device addresses and select codes have been set (section 1).
3) Verify that the devices are properly interconnected (section 2).
4) Vent the vacuum chamber and open the front door.
5) Bolt the clock to the TED.
6) Bolt the TED to the chamber base plate.
7) Connect the power, analog monitor, and RF leads to the clock.
8) Connect power lines to the TED.
9) Connect the cable to the thermistor probe receptacle box.
10) Plug a thermistor into jack 0 and mount it to the TED. This thermistor MUST be in place.
11) Plug a thermistor into jack 1 and mount it to the chamber base plate. This thermistor MUST be in place as well.
12) Plug additional thermistors into the remaining jacks as necessary, being sure to record where they are placed so that the INIT file can be edited properly.
13) Close the chamber door and start the vacuum system.
14) When the pressure is below 100 millitorr, turn duplex power on.
15) Start the clock and verify that it is working. Check the counter frequency reading and the analog monitors (section 3).
16) Check the reference signal coming in from room 204.
17) Place a system disk in drive 0.
18) Place a data storage disk in drive 1.
19) Locate a Softcopy Graphics Tape and POWER FAIL TAPE.
20) Setup the initialization file by running INIT (section 5).
21) Run GATHER.

7.0 Obtaining Data During a Test

Before producing lists or plots on the plotter/printer, see if the print head is at the top of the page, and check the page boundaries. Replace the paper if the margin is colored red. Check the calculator printer from time to time, and replace the paper if it is low. A calculator printer error (eg, printer out of paper) will halt program execution.

Table II shows the subroutines called in when the twelve special function keys are pressed. Additional subroutines can be requested before the last one has finished running. Three subroutines require user input: .PL005, .PL013, and .SCA06. Instructions for these are printed on the calculator printer during execution. They require the selection of additional special function keys.

Table II: Subroutines

special function key	subroutine name	output device	description
f0	.INI18	718	Lists the contents of the INIT file on the display terminal.
f1	.INT18	718	Plots time interval average measurements as a function of time on the display terminal.
f2	.OFF18	718	Plots fractional frequency offset as a function of time on the display terminal.
f3	.SIG18	718	Plots Allan Variance as a function of averaging time on the display terminal. The values of the plotted points are printed on the calculator printer.
f4	.PLO18	718	Plots scanner channel data as a function of time on the display terminal. This subroutine requires selection of a channel number. A channel number and name are displayed on the calculator. Special function key f0 increments the count, f1 decrements the count, and f2 plots the displayed channel.
f5	.CUI16	16	Produces a calculator printer listing of key status variables.
f6	.INI05	705	Lists the contents of the INIT file on the plotter/printer.
f7	.INT05	705	Plots time interval average measurements as a function of time on the plotter/printer.
f8	.OFF05	705	Plots fractional frequency offset as a function of time on the plotter/printer.
f9	.SIG05	705	Plots Allan Variance as a function of averaging time on the plotter/printer. The values of the plotted points are printed on the calculator printer.
f10	.PLO05	705	Plots scanner channel data as a function of time on the plotter/printer. This subroutine requires selection of a channel number. A channel number and name are displayed on the calculator. Special function key f0 increments the count, f1 decrements the count, and f2 plots the displayed channel.
f11	.SCA06	706	Produces a plotter/printer listing of data collected through the scanner. This subroutine requires an additional entry. Press special function key f0 to list all the data, and f1 to list data collected since midnight.

If the calculator error halts for any reason while a subroutine is executing, turn the calculator off and back on, or enter:

ldp <execute>

This will load in the REBOOT program, and program execution will pick up where it left off.

If an error halts the program, and the program is simply restarted in the wait loop, the interrupts will no longer be functional and GATHER will not collect data.

8.0 Obtaining Data Once a Test is Through

This section describes the various forms of hard copy that can be obtained using system programs. Each block contains the program file name, the required inputs, a description of the program, and the name of the display terminal version, if one exists.

program: INIT
input: default or variable file, list, printer
screen version: same
description: lists the contents of either the INIT.D or INIT.V files

Precision Oscillator Test Facility

Program: INIT
Written: 12-17-80
For: Space Applications Branch
US Naval Research Lab

Revision Date: 01-11-80
Measurement Interval: 0d 1h 00m 0s
Counter Gate Time: 100 msec

**** Temperature Control ****

##	Temperature (deg C)	Duration Days	Hrs	Mins	Secs
**	**********	****	***	****	****
1	25	1	0	0	0

**** Scanner Control ****

##	Channel Number	Channel Name	Channel Function
**	******	******	********
1	0	Control	DC V
2	1	Beam I	DC V
3	2	C Field	DC V
4	3	Ionizer	DC V
5	4	Cs Oven	DC V
6	5	Ion Pump	DC V
7	6	Qtz Oven	DC V
8	7	EM	DC V
9	8	DV Voltage	DC V
10	9	DC Current	DC V
11	10	RF Power	DC V
12	11	Tmp Offset	DC V
13	12	Vacuum	DC V
14	13	Temp	DC V

program: SCA06
input: data file name, start date, additional columns
screen version: SCA13
description: lists the contents of test data files from the specified start date. Additional columns can be generated from those which already exist, as for example, POWER = SUPPLY V x SUPPLY I.

***** Precision Oscillator Test Facility *****

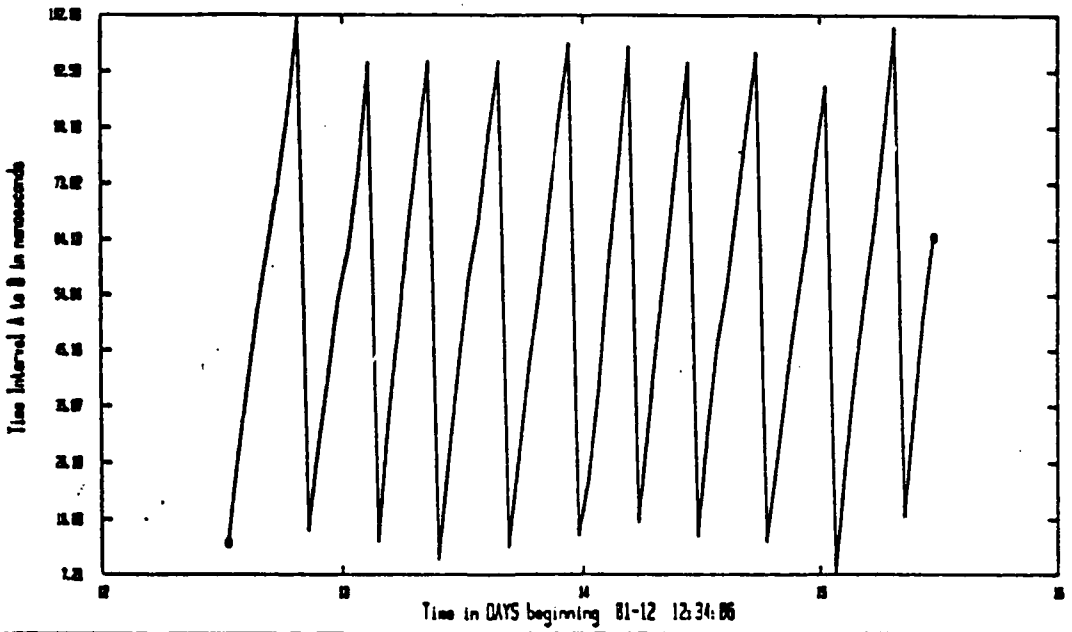
Start Date of Test:	12-19-83
Signal Frequency:	10.23 MHz
Gate Time:	100 msec
Number of Points:	48
Measurement Interval:	1800 secs
Data File Name:	K5#10A
Program:	SCA06

Date	Time	Phase	Supply I	Supply V	Ionizer	trash	Qtz Ovn
12-19	16:00:28	2.5390E-08	1.315	2.563	1.136	1.261	0.767
12-19	16:30:28	8.6680E-08	1.314	2.563	1.136	1.282	0.767
12-19	17:00:28	7.2590E-08	1.315	2.564	1.136	1.281	0.767
12-19	17:30:28	5.7940E-08	1.315	2.564	1.136	1.266	0.767
12-19	18:00:28	4.5400E-08	1.316	2.564	1.136	1.269	0.767
12-19	18:30:28	3.1070E-08	1.313	2.564	1.133	1.256	0.765
12-19	19:00:28	1.8970E-08	1.311	2.564	1.131	1.234	0.757
12-19	19:30:28	1.0129E-07	1.302	2.564	1.123	1.238	0.745
12-19	20:00:28	8.7720E-08	1.293	2.564	1.117	1.266	0.732
12-19	20:30:28	7.2870E-08	1.288	2.564	1.115	1.228	0.720
12-19	21:00:28	5.4710E-08	1.283	2.564	1.114	1.243	0.710
12-19	21:30:28	3.7720E-08	1.281	2.564	1.114	1.218	0.702
12-19	22:00:28	2.2650E-08	1.278	2.564	1.114	1.258	0.696
12-19	22:30:28	1.0184E-07	1.276	2.564	1.115	1.260	0.691
12-19	23:00:28	8.4120E-08	1.275	2.564	1.115	1.248	0.687
12-19	23:30:28	6.9550E-08	1.275	2.564	1.115	1.213	0.684
12-20	00:00:28	5.2770E-08	1.274	2.564	1.115	1.213	0.682
12-20	00:30:28	3.4370E-08	1.272	2.564	1.115	1.261	0.680
12-20	01:00:28	1.6860E-08	1.272	2.565	1.115	1.269	0.679
12-20	01:30:28	9.5890E-08	1.272	2.565	1.115	1.250	0.678
12-20	02:00:28	7.8600E-08	1.272	2.565	1.115	1.218	0.677
12-20	02:30:28	6.0700E-08	1.271	2.565	1.115	1.237	0.676
12-20	03:00:28	4.4120E-08	1.270	2.565	1.115	1.262	0.676
12-20	03:30:28	2.6490E-08	1.270	2.565	1.115	1.219	0.675
12-20	04:00:28	9.7600E-09	1.271	2.565	1.115	1.240	0.675
12-20	04:30:28	9.3490E-08	1.271	2.565	1.115	1.209	0.675
12-20	05:00:28	7.6020E-08	1.270	2.565	1.115	1.215	0.674
12-20	05:30:28	5.7860E-08	1.270	2.565	1.115	1.245	0.674
12-20	06:00:28	4.2630E-08	1.270	2.565	1.115	1.216	0.674
12-20	06:30:28	2.4910E-08	1.271	2.565	1.115	1.226	0.674
12-20	07:00:28	7.7600E-09	1.270	2.565	1.115	1.250	0.674
12-20	07:30:28	8.5540E-08	1.270	2.565	1.115	1.258	0.674
12-20	08:00:28	6.9420E-08	1.270	2.565	1.115	1.237	0.674
12-20	08:30:28	5.0420E-08	1.271	2.565	1.115	1.193	0.674
12-20	09:00:28	3.2570E-08	1.270	2.565	1.115	1.240	0.674
12-20	09:30:28	1.5900E-08	1.270	2.565	1.116	1.244	0.677
12-20	10:00:28	9.5210E-08	1.273	2.565	1.118	1.234	0.687
12-20	10:30:28	7.9630E-08	1.278	2.565	1.122	1.258	0.701
12-20	11:00:28	6.1060E-08	1.284	2.565	1.127	1.254	0.713
12-20	11:30:28	4.7290E-08	1.290	2.565	1.131	1.254	0.723
12-20	12:00:28	3.1170E-08	1.294	2.565	1.131	1.287	0.731

program: QUI16
input: data file name
screen version: none
description: produces a quick list of key variables

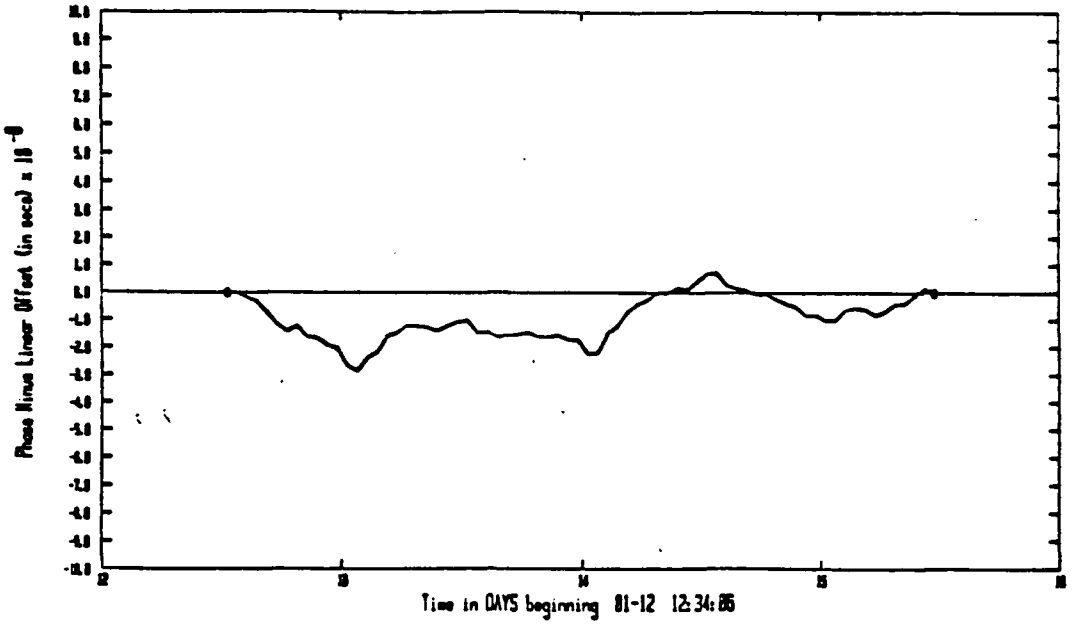
```
================
Gate Time in
msec:      100.00
================
Total Number of
Points
Collected:  1407
================
Time Interval
Between
Measurements
in secs:      900
================
Date:   01-16-83
================
Signal Period:
    9.775175e-08
================
Voltage Channels
Sampled:       12
================
```

program: INT05
input: data file name, start date, end date
screen version: INT18
description: plots counter time interval average measurements as a function of time



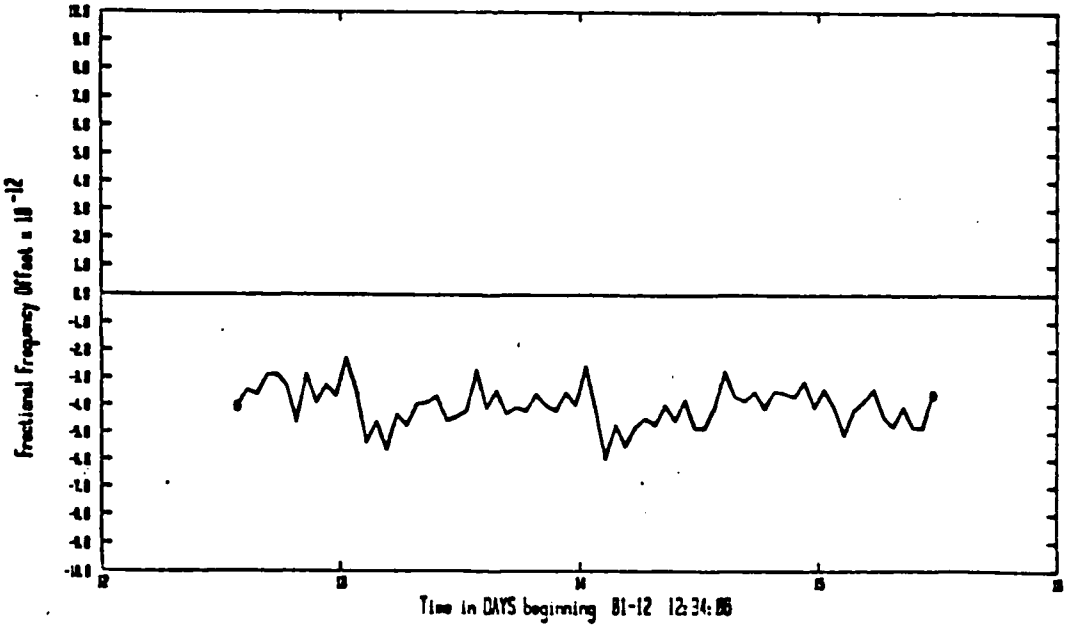
Date	Program	Data File Name	Points Plotted	Frequency	Measurement Interval	Counter Gate Time
81-12-81	Time Interval Plot	ppm#6	72	10.23 MHz	3628 secs	10 msec

program: PHA05
input: data file name, start date, end date
screen version: none
description: plots cumulative phase minus linear offset (line drawn between endpoints) as a function of time



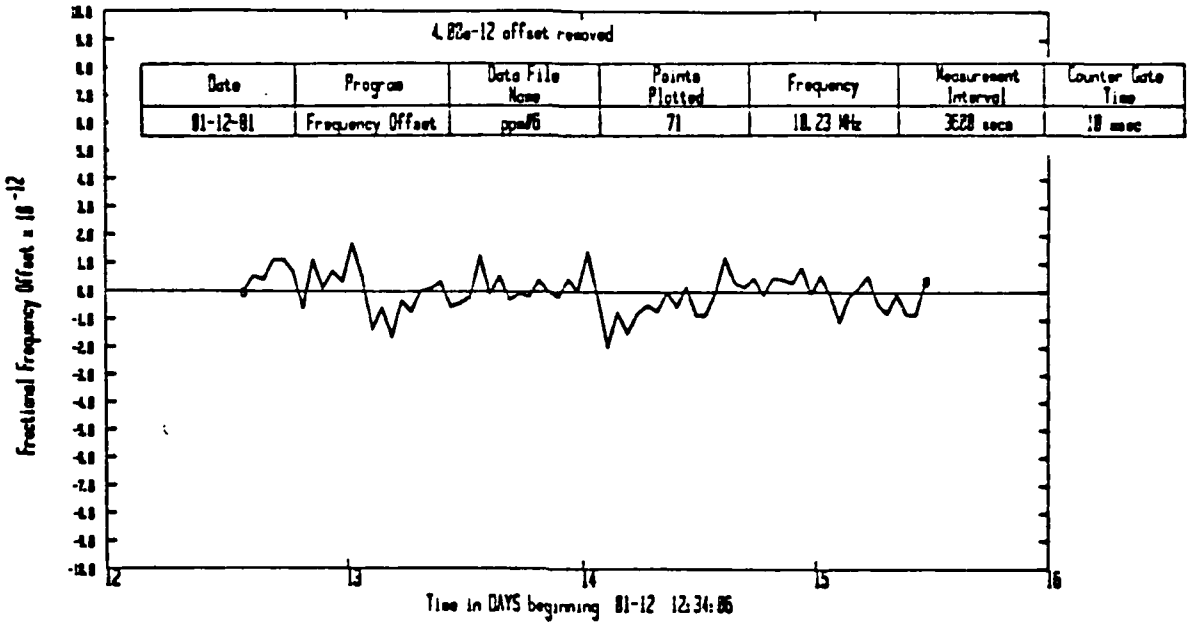
Date	Program	Data File Name	Points Plotted	Frequency	Measurement Interval	Counter Gate Time
81-12-01	Cumulative Phase	ppm86	71	10.23 MHz	3528 secs	10 msec

program: OFF05
input: data file name, start date, end date
screen version: OFF10
description: plots fractional frequency offset (delta F/F) as a function of time

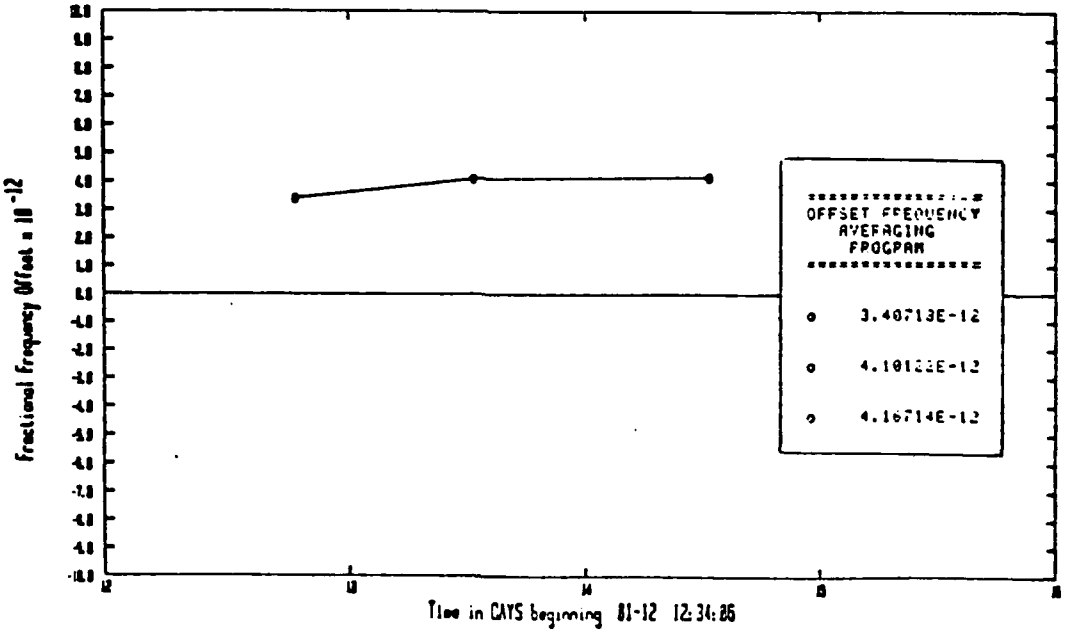


Date	Program	Data File Name	Points Plotted	Frequency	Measurement Interval	Counter Gate Time
81-12-01	Frequency Offset	ppm86	71	10.23 MHz	3528 secs	10 msec

program: VOFF05
input: data file name, start date, end date, offset to remove
screen version: VOFF19
description: similar to OFF05, but contains option for entering a constant offset to be removed. In the example, a 4e-12 offset was removed. See the OFF05 graph for the difference.

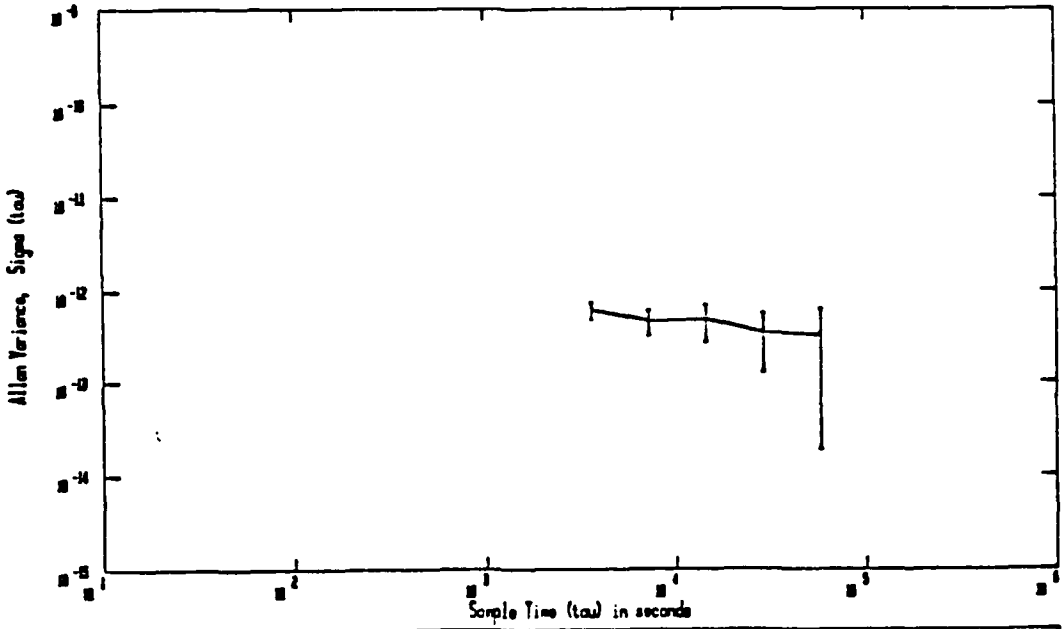


program: AVG05
input: data file name, start average, end average
screen version: none
description: similar to OFF05, but allows data over specified intervals to be averaged. The data is displayed at the midpoint of the interval averaged on the graph. This program is useful for calculating temperature coefficients given a temperature profile.



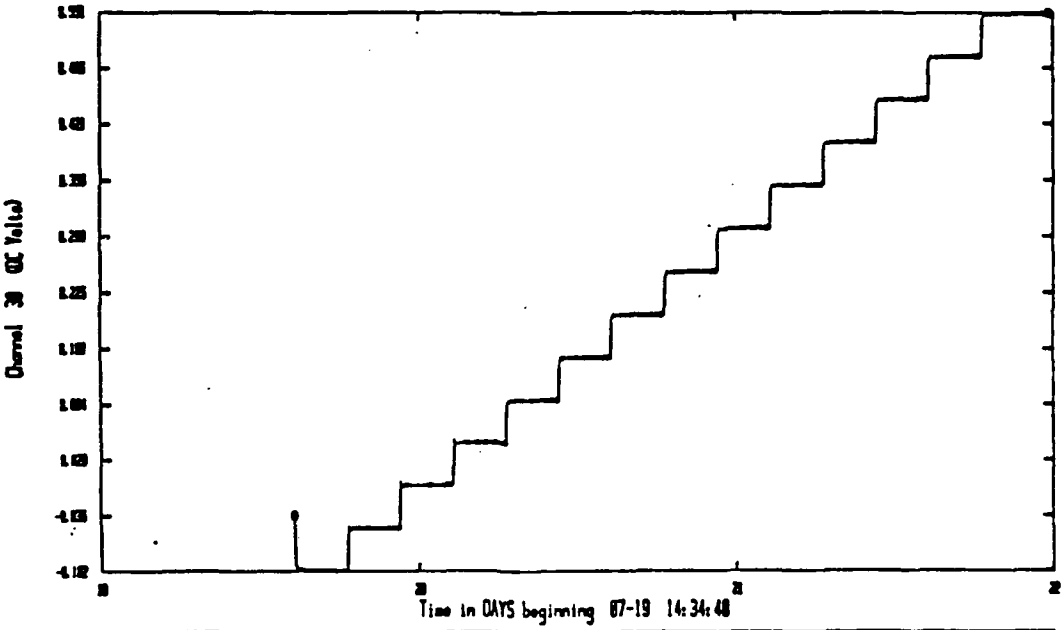
Date	Program	Data File Name	Points Plotted	Frequency	Measurement Interval	Counter Gate Time
81-12-81	Offset Freq Ave	ppm05	3	10.23 MHz	3528 secs	10 msec

program: SIG05
input: data file name, start date, end date
screen version: SIG19
description: plots Allan variance as a function of averaging time



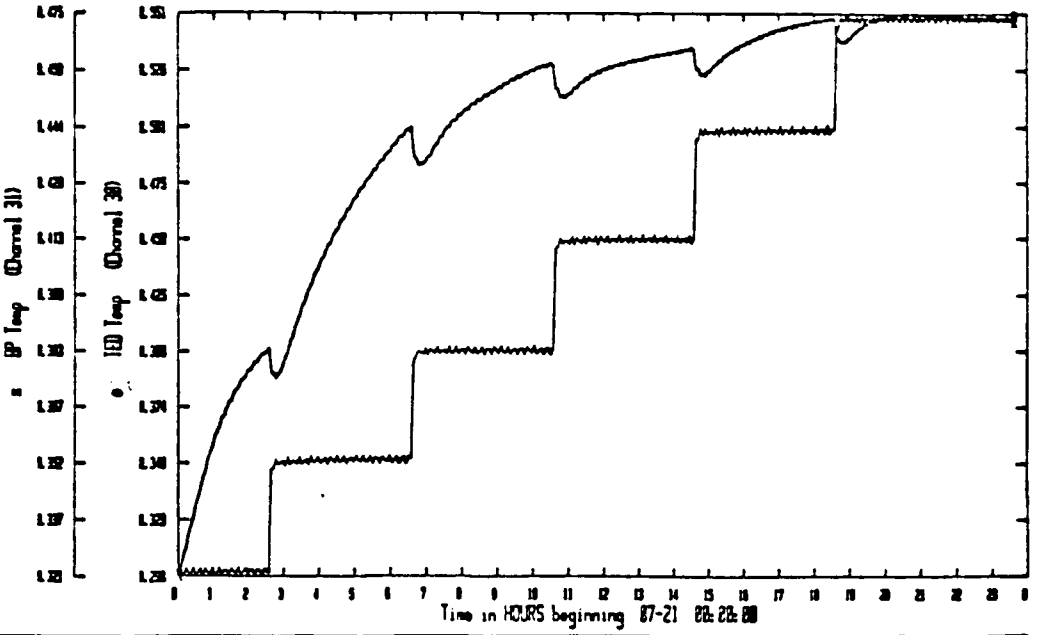
Date	Program	Data File Name	Points Found	Frequency	Measurement Interval	Counter Gate Time
01-12-81	Allan Variance	ppm86	71	10.23 MHz	3628 secs	10 msec

program: PL005
input: data file name, start date, end date, channel number
screen version: PL018
description: plots scanner voltage data as a function of time



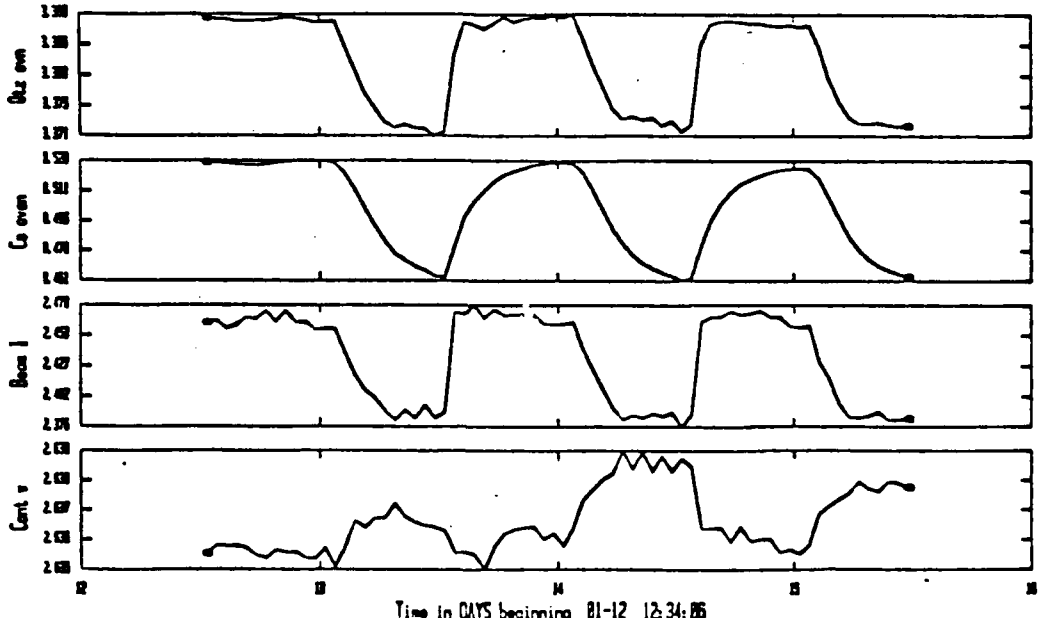
Date	Program	Data File Name	Points Plotted	Frequency	Measurement Interval	Channel Name
07-19-83	Channel Plot	test	857	10.00 MHz	240 secs	TED Temp

program: MLT05
input: data file name, start date, end date, channel numbers
screen version: MLT18
description: similar to PL005, but plots up to 4 lines on the same graph. Enter <cont> to plot once the last channel has been entered.



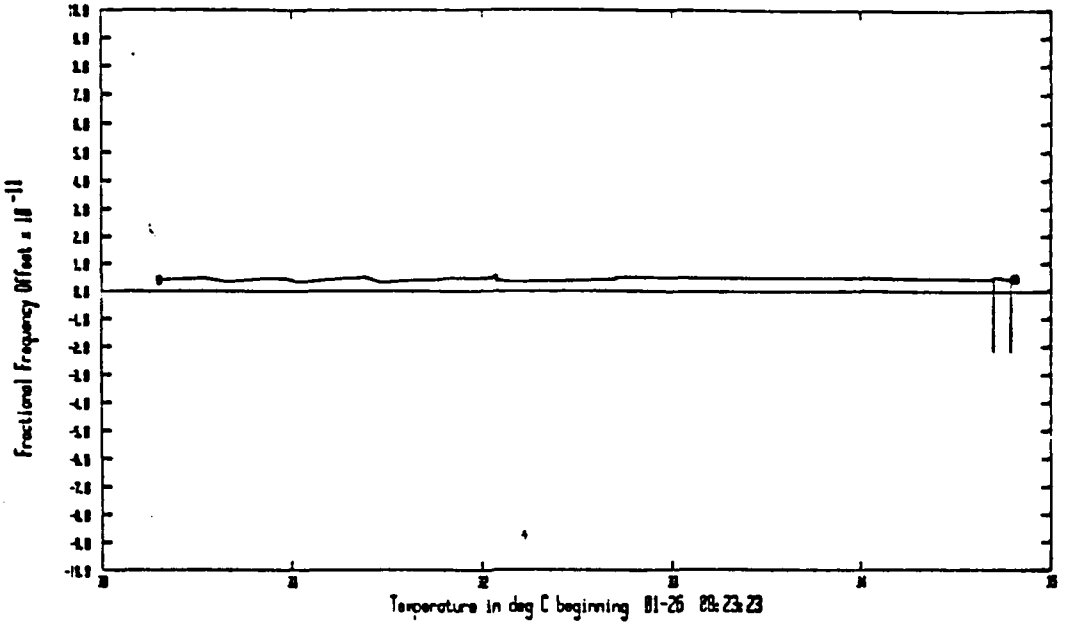
Date	Program	Data File Name	Points Plotted	Frequency	Measurement Interval	Channel Function
87-19-83	Analog Multiplot	test	355	18.88 MHz	248 secs	DC Voltage

program: STK05
input: data file name, start date, end date, channel numbers
screen version: STK18
description: similar to PL005, but stacks up to 4 plots, one on top of the other. Enter <cont> to plot once the last channel has been entered.



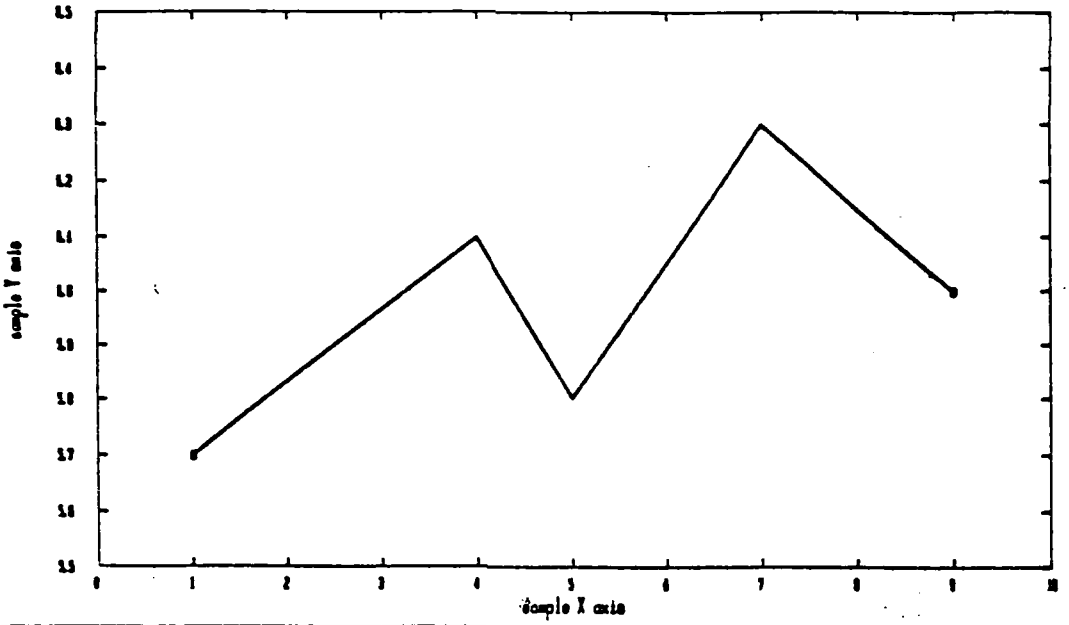
Date	Program	Data File Name	Points Plotted	Frequency	Measurement Interval	Channel Function
81-12-81	Analog Stack Plot	ppnd6	72	18.23 MHz	3528 secs	DC Voltage

program: FVT05
input: data file name, temperature channel number, temperature scale factor, start date, end date, lower limit, upper limit
screen version: none
description: plots fractional frequency offset as a function of temperature. Delta F/F values are arranged in order of ascending temperature, regardless of the order in which temperatures were obtained.



Date	Program	Data File Name	Points Plotted	Frequency	Measurement Interval	Counter Gate Time
81-26-81	Freq Offset vs Tmp	ppm#8	259	10.23 MHz	3600 secs	10 msec

program: USR05
input: X min, X max, number X axis tic marks, number of X axis decimal places, X axis label, Y min, Y max, number of Y axis tic marks, number of Y axis decimal places, Y axis label, X1, Y1, X2, Y2, X3, Y3, etc
screen version: none
description: a general purpose user-entry plot routine



Date	Program	Data Source	Points Plotted	Clock	Associated Data File	Executed by
07-29-83	Gen Purpose Plot	User Entry	5	N.A.	N.A.	Kevin M. Miller

9.0 Program Library

Figure 9.1 shows a catalog listing of the programs on the system disk. The file name is followed by a character which indicates the file type. P = program, D = data, M = memory, and K = keys. The next three columns contain data which indicate the number of records, the beginning track number, and the beginning record number.

```
CAT DRIVE    0 /A
AVL RCRDS 961
NAME TYPE SIZE      #REC TRCK RCRD
- - - - - - - - - - - - - - - - -
INIT.D D            30   T6   R16
MEMORY M  238628    94   T39  R10
INT10  P  47448     10   T16  R2
OFF05  P  47608     19   T14  R6
OFF10  P  47448     19   T7   R16
MAGPRG P  19548     8    T43  R23
SIG10  P  40988     17   T8   R25
GATHER P  65028     26   T25  R26
FVT05  P  49868     20   T9   R12
INIT   P  140108    55   T20  R23
INT05  P  44528     10   T10  R2
PLO10  P  49548     20   T8   R5
PLO05  P  49488     20   T14  R25
.INT05 P  37308     15   T24  R2
SCA06  P  43328     17   T15  R15
REBOOT P  14308     6    T24  R28
SIG05  P  42528     17   T16  R20
.QUI16 P  8168      4    T19  R10
QUI16  P  9588      4    T10  R13
KEYS   K  968       1    T26  R22
PHA05  P  47688     19   T26  R23
.OFF05 P  39928     16   T10  R17
AVG05  P  49628     20   T27  R12
USR05  P  49328     20   T28  R2
MLT05  P  58608     23   T28  R22
STK05  P  61068     24   T29  R15
.SIG05 P  36288     15   T19  R3
MAGDAT P  22748     9    T43  R14
INIT.V D            30   T19  R23
VOFF05 P  49548     20   T32  R9
.PLO05 P  44768     10   T17  R25
.SCA06 P  29568     12   T13  R24
VOFF10 P  49388     20   T32  R29
STK10  P  60848     24   T38  R12
SCREEN K  2220      1    T19  R22
.INI10 P  37808     15   T23  R17
.INT10 P  37968     15   T22  R10
.OFF10 P  40388     16   T6   R8
MLT10  P  58988     24   T33  R21
.SIG10 P  35848     14   T23  R3
.PLO10 P  44968     10   T17  R7
.INI05 P  26188     11   T24  R17
SCA10  P  54428     22   T42  R22
PATCH  P  10608     5    T25  R4
```

Figure 9.1

Beginning on the next page is a short description of the files not documented elsewhere in this paper. Following that is a line printer listing of each of the programs contained on the disk.

File Descriptions

MAGPRG - is a program that copies programs onto magnetic tape. The program requires an Ideas 4600 series tape controller on bus 7 set to device address 16. The tapes produced are PIP-readable ANSI standard format. The program loads files in at line 100 to prevent skewing of the margins, so the blank lines on the program listing are necessary. To remove the line numbers in an editor, delete the first three columns.

KEYS - is a key file that contains *sfg statements. This file is called by GATHER to enable the keyboard execution of subroutines. The asterix on a special function command is the same as <execute>.

MAGDAT - is a program that copies test file data onto magnetic tape. The program requires an Ideas 4600 series tape controller on bus 7 set to device address 16. The tapes produced are PIP-readable ANSI standard format.

SCREEN - is a keys file that loads special function keys for listing programs and catalogs on the display screen. The Softcopy graphics Library binary program must also be loaded. The keys have the following functions:

f0 sets the program line counter to 0
f1 displays the next 35 lines of program
f2 clears the screen
f3 displays a catalog of drive 0
f4 displays a catalog of drive 1

PATCH - takes files that have been prematurely closed because of an interrupted test, and patches them up so they can be read by data handling routines. This is a program worth knowing about.

The pages that follow contain a line printer listing of each of the system programs.

Table III

##	file	page	##	file	page	##	file	page
1	INIT	45	14	.OFF05	69	27	.PL018	93
2	GATHER	49	15	.OFF18	71	28	MLT05	95
3	REBOOT	51	16	VOFF05	73	29	MLT18	97
4	PATCH	52	17	VOFF18	75	30	STK05	99
5	.INI05	53	18	AVG05	77	31	STK18	101
6	.INI18	54	19	FVT05	79	32	QUI16	103
7	INT05	55	20	SIG05	81	33	.QUI16	104
8	INT18	57	21	SIG18	83	34	SCA06	105
9	.INT05	59	22	.SIG05	85	35	SCA18	107
10	.INT18	61	23	.SIG18	86	36	.SCA06	109
11	PHA05	63	24	PL005	87	37	USR05	110
12	OFF05	65	25	PL018	89	38	MAGPRG	112
13	OFF18	67	26	.PL005	91	39	MAGDAT	114

9.1 Program: INIT

```
0: "PROGRAM,  DATA COLLECTION INITIALIZATION ROUTINE";
1:
2: "*************************************************";
3: "*  Kevin Miller                 *  File: INIT      *";
4: "*  December 28, 1980            *  Update: July 18, 1983  *";
5: "*  Naval Research Laboratory    *  Output Device: 7245, 1350  *";
6: "*  Space Applications Branch    *  Select Code: 705, 718  *";
7: "*************************************************";
8:
9: dsp "Insert Softcopy Graphics Tape";stp
10: ldb 1;dev "PRINTER",706,"PLOTTER",705,"SCREEN",718,"CLOCK",9
11: dim A[25],A$[25,6],B[25],B$[25,10],C$[25,1],V$[4]
12: dim M$[70],T$[14],R$[15],S$[4,2],Y$[2],D$[6],I$[8]
13: dim A,B,C,F,G,I,K,L,M,Z
14: "st")S$[1],"nd")S$[2],"rd")S$[3],"th")S$[4]
15: files INIT.V:0,INIT.D:0;fzd 0
16: ent "Variable or Default Init File",R$;if flg13;jmp 0
17: 1)F;if cap(R$[1,1])-"D";2)F;jmp 2
18: if cap(R$[1,1])#"Y";dsp "Invalid Entry";wait 1500;jmp -2
19: "START";ent "Initialize, List, Gather or End",R$;if flg13;jmp 0
20: cap(R$[1,1])R$;if R$-"G";drive 0;get "GATHER"
21: if R$-"E";end
22: if R$#"I" and R$#"L";dsp "Invalid Entry";wait 1500;jmp -3
23: rread F,1,D$,A,B,I$,G
24: rread F,2,A[1],A$[1];for L-2 to A;sread F,A[L],A$[L];next L
25: rread F,4,B[1],B$[1],C$[1];for L-2 to B;sread F,B[L],B$[L],C$[L];next L
26: cap(R$[1,1])R$;if R$-"L";gsb "LIST"
27:
28: "INITIALIZE";
29: wrt "CLOCK","R";red "CLOCK",T$;T$[1,2]&T$[4,5]&T$
30: ent "Current Year:   YY",Y$;if flg13;jmp 0
31: if val(Y$)<80;dsp "Too Small";wait 1500;jmp -1
32: if val(Y$)>90;dsp "Too Large";wait 1500;jmp -2
33: T$&Y$)D$;"")R$;ent "Print Instructions",R$;if flg13;jmp 0
34: if cap(R$[1,1])-"N";gto "Vacuum Temp Control"
35: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
36: spc 3;prt "***************","      FILE"," INITIALIZATION"
37: prt "***************";spc 2;prt "Four parameters"
38: prt "are initialized","by this program:"," ","1) Vacuum"
39: prt "     Temperature","     Control"
40: prt " 2) Voltage","     Recording","     Channels"," 3) Measurement"
41: prt "     Interval"," 4) Counter Gate","     Time";spc 2
42: prt "With the program","you can","selectively edit","any or all"
43: prt "portions of the","init file. Enter","CONTINUE in"
44: prt "entry mode to","proceed from one","section of the"
45: prt "program to the","next.";spc 3
46:
47: "Vacuum Temp Control";
48: "")R$;ent "Edit Temperature Control",R$
49: if cap(R$[1,1])-"N";gto "Voltage Channels"
50: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
51: "T Again","")R$;ent "List Present Temp Setpoints",R$
52: if cap(R$[1,1])-"N";jmp 3
53: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
54: gsb "List T"
55: "All T","")R$;ent "Change All Entries",R$
56: if cap(R$[1,1])-"N";gto "Add T"
57: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
58: 25)A[1,0)K;"")V$;for L-1 to 25;gsb "Enter T"
59: if K and L-1;0)K;jmp 2
60: if K;gto "T Set"
61: L)A;next L;25)A
62: dsp "25 Setpoint Max Reached";wait 2000;gto "T Set"
63: "Add T";if A=25;gto "Delete T"
64: "")R$;ent "Add Temperature Setpoints",R$
65: if cap(R$[1,1])-"N";gto "Delete T"
66: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
67: 0)K;"")V$;for L-A+1 to 25;gsb "Enter T"
68: if K;gto "Delete T"
69: L)A;next L;25)A
70: dsp "25 Setpoint Max Reached";wait 2000
71: "Delete T";if A-1;gto "Insert T"
72: "")R$;ent "Delete Temperature Setpoints",R$
73: if cap(R$[1,1])-"N";gto "Insert T"
74: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
75: if A-1;dsp "Cannot Delete Last Temp Setpoint";wait 2000;gto "Insert T"
76: ent "Entry to Delete",D;if flg13 and A<25;gto "Insert T"
77: if flg13 and A=25;gto "Modify T"
78: if D<1 or D>A;dsp "Out of Range";wait 1500;gto -2
79: if D-A;dsp str(D)&" Deleted";wait 2000;jmp 2
80: dsp "Renumbering"&str(D+1)&" thru"&str(A);wait 2000
81: for L-D to A-1;A[L+1])A[L];A$[L+1])A$[L];next L;A-1)A;gto -6
82: "Insert T";if A=25;gto "Modify T"
83: "")R$;ent "Insert Temperature Setpoints",R$
84: if cap(R$[1,1])-"N";gto "Modify T"
85: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
86: ent "Insert Before Which Entry",Z;int(Z)Z;if flg13;gto "Modify T"
```

```
87: if Z<1 or Z>A;dsp "Out of Range";wait 1500;jmp -1
88: for L=A-1 to Z+1 by -1;A[L-1]}A[L];A$[L-1]}A$[L];next L
89: 0}K;" New"}V$;Z}L;gsb "Enter T"
90: if K;jmp -1
91: A-1}A;if A<25;gto -5
92: dsp "25 Setpoint Max Reached";wait 2000
93: "Modify T:","")R$;ent "Modify Individual Entries",R$
94: if cap(R$[1,1])="N";gto "T Bot"
95: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
96: ent "Entry to Change",L;if flg13;gto "T Bot"
97: if L<1 or L>A;dsp "Out of Range";wait 1500;jmp -1
98: 0}K;" New"}V$;gsb "Enter T"
99: if K;jmp -1
100: gto -4
101: "T Bot":rprt F,1,D$,A,B,I$,100;rprt F,2,A[1],A$[1]
102: for L=2 to A;sprt F,A[L],A$[L];next L
103: "")R$;ent "List New Temperature Setpoints",R$
104: if cap(R$[1,1])="N";jmp 3
105: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
106: gsb "List T"
107: "")R$;ent "Pass thru Temp Routine Again",R$
108: if cap(R$[1,1])="Y";gto "T Again"
109: if cap(R$[1,1])#"N";dsp "Answer YES or NO";wait 1500;jmp -2
110: gto "Voltage Channels"
111: "List T":sme ;prt "Temp Setpoints:","  ","   Temp Duration"
112: prt "## degC (DDHHMM)","-- ---- ---------"
113: fxd 0;for L=1 to A;str(A[L])M$;if abs(A[L])<10;" "&M$}M$
114: if L#A;str(L)&" "&M$&"    "&A$[L])M$;if L>9;M$[2])M$
115: if L=A;str(L)&" "&M$&"    to end")M$;if L>9;M$[2])M$
116: prt M$;next L;spe 3;ret
117: "Enter T";4(Lmod10=0)+(Lmod10=1)+2(Lmod10=2)+3(Lmod10=3)+4(Lmod10>3)}I
118: if L>9 and L<14;4}I
119: dsp V$&str(L)&S$[I]&" Temperature Setpoint";ent "",A[L]
120: if flg13 and L=1;dsp "25 deg C Default";wait 1500
121: if flg13 and L#1;1}K;ret
122: if A[L]>-50 and A[L]<-99;jmp 2
123: dsp "Specify from -50 to -99 deg C";wait 2000;jmp -4
124: dsp "Time at"&str(L)&S$[I]&" Setpoint:  DDHHMM";ent "",T$
125: if flg13;dsp "Time Must be Specified";wait 1500;jmp -1
126: if len(T$)#6;dsp "Not in Proper Form";wait 1500;jmp -2
127: T$}A$[L];ret
128:
129: "Voltage Channels":
130: "")R$;ent "Edit Voltage Recording Channels",R$
131: if cap(R$[1,1])="N";gto "Measurement Interval"
132: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
133: "VC Again","")R$;ent "List Present Channel Settings",R$
134: if cap(R$[1,1])="N";gto "All VC"
135: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
136: gsb "List VC"
137: "All VC","")R$;ent "Change All Entries",R$
138: if cap(R$[1,1])="N";gto "Add VC"
139: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
140: 0}K}B;"")V$;for L=1 to 25;gsb "Enter VC"
141: if K;gto "VC Bot"
142: L}B;next L;25}B
143: dsp "25 Channel Max Reached";wait 2000;gto "VC Bot"
144: "Add VC":if B=25;gto "Delete VC"
145: "")R$;ent "Add Voltage Channels",R$
146: if cap(R$[1,1])="N";gto "Delete VC"
147: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
148: B}K;"")V$;for L=B+1 to 25;gsb "Enter VC"
149: if K;gto "Delete VC"
150: L}B;next L;25}B
151: dsp "25 Channel Max Reached";wait 2000
152: "Delete VC":if B=0;gsb "Modify VC"
153: "")R$;ent "Delete Voltage Channel Entries",R$
154: if cap(R$[1,1])="N";gto "Modify VC"
155: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
156: if B=0;dsp "All Channels Deleted";wait 2000;gto "VC Bot"
157: ent "Entry to Delete",D;if flg13;gto "Modify VC"
158: if D<1 or D>B;dsp "Out of Range";wait 1500;jmp -2
159: if D=B or B=1;dsp str(D)&" Deleted";wait 1500;jmp 2
160: dsp "Renumbering"&str(D+1)&" thru"&str(B);wait 1500
161: for L=D to B-1;B[L+1])B[L];B$[L+1])B$[L];C$[L+1])C$[L];next L
162: B-1)B;jmp -6
163: "Modify VC":if B=0;gto "VC Bot"
164: "")R$;ent "Modify Individual Entries",R$
165: if cap(R$[1,1])="N";gto "VC Bot"
166: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
167: ent "Entry to Change",L;if flg13;gto "VC Bot"
168: if L<0 or L>B;dsp "Out of Range";wait 1500;jmp -1
169: 0}K;" New"}V$;gsb "Enter VC"
170: if K;jmp -1
171: gto -4
172: "VC Bot":rprt F,1,D$,A,B,I$,100;rprt F,4,B[1],B$[1],C$[1]
173: for L=2 to B;sprt F,B[L],B$[L],C$[L];next L
174: "")R$;ent "List New Channel Settings",R$
```

```
177: gsb "List VC"
178: "")R$;ent "Pass thru Channel Routine Again",R$
179: if cap(R$[1,1])-"Y";gto "VC Again"
180: if cap(R$[1,1])#"N";dsp "Answer YES or NO";wait 1500;jmp -2
181: gto "Measurement Interval"
182: "List VC":spc ;prt "Voltage Channel","Settings:",,"
183: prt "## Chan Function","as aaaa aaaaaaaa"
184: fxd 0;for L=1 to B;str(B[L])M$;if B[L]<10;" "&M$)M$
185: str(L)&" "&M$&"  ")M$;if L>9;M$[2]M$
186: if C$[L]="D";M$&"DC Volts")M$
187: if C$[L]="A";M$&"AC Volts")M$
188: if C$[L]="O";M$&"Ohms")M$
189: prt M$;next L;spc 2;prt "##       Name","as    aaaaaaaaaa"
190: for L=1 to B;str(L)&"    "&B$[L])M$;if L>9;M$[2]M$
191: prt M$;next L;spc 3;ret
192: "Enter VC":if L>9 and L<14;4)1;jmp 2
193: 4(Lmod10=0)+(Lmod10=1)+2(Lmod10=2)+3(Lmod10=3)+4(Lmod10>3))1
194: dsp V$&str(L)&S$[1]&" Channel Number";ent "",B[L]
195: if flg13;1)K;ret
196: if B[L])=0 and B[L]<=39;jmp 2
197: dsp "Channels are numbered 0 thru 39";wait 2000;jmp -3
198: dsp V$&str(L)&S$[1]&" Channel Name";ent "",T$
199: if flg13;"Channel"&str(L-1))T$;dsp T$;wait 1000
200: if len(T$))10;dsp "Name Too Long";wait 1500;jmp -2
201: T$)B$[L];"")R$;ent "DC Volts, AC Volts or Ohms",R$
202: cap(R$[1,1]))R$;if flg13;"O")R$
203: if R$#"D" and R$#"A" and R$#"O";dsp "Invalid Entry";wait 1500;jmp -2
204: R$)C$[L];if R$="D";dsp " DC Volts";wait 1000
205: if R$="A";dsp " AC Volts";wait 1000
206: if R$="O";dsp " Ohms";wait 1000
207: ret
208:
209: "Measurement Interval":
210: "")R$;ent "Change Measurement Interval",R$
211: if cap(R$[1,1])="N";gto "START"
212: if cap(R$[1,1])#"Y";dsp "Answer YES or NO";wait 1500;jmp -2
213: spc 3;prt "Present","Measurement","Interval:","  ","    "&I$;spc 2
214: "")T$;ent "Measurement Interval:  DDHHMMSS",T$;if flg13;jmp 0
215: if len(T$)#8;dsp "Not in Proper Form";wait 1500;jmp -1
216: T$)I$;prt "New","Measurement","Interval:","  ","    "&I$;spc 3
217: rprt F,1,D$,A,B,I$,100;gto "START"
218:
219: "LIST":
220: "")R$;ent "List on Screen or Printer",R$
221: if cap(R$[1,1])="P";gto "Printer"
222: if cap(R$[1,1])#"S";dsp "Answer SCREEN or PRINTER";wait 1500;jmp -2
223:
224: "Screen":
225: fxd 0;psc 718;hdcpy 0;pclr
226: scl 0,2,0,20;xax 0;xax 15;xax 16;xax 19;xax 20
227: yax 0;yax 1,0,0,16;yax 2
228: wtb "SCREEN",3,"em::sn::sz::um::pe0,::pa287,967;::"
229: wtb "SCREEN","pe1,::cs,::tx    Thermal Vacuum Test Facility   ",3,10
230: wtb "SCREEN","pe0,::pa 30,920;::pe1,::tx Program: INIT",3,10
231: wtb "SCREEN","pe0,::pa 30,890;::pe1,::tx Written: 12-17-80",3,10
232: wtb "SCREEN","pe0,::pa 30,860;::pe1,::"
233: wtb "SCREEN","tx For:    Space Applications Branch",3,10
234: wtb "SCREEN","pe0,::pa 30,830;::pe1,::"
235: wtb "SCREEN","tx          US Naval Research Lab",3,10
236: wtb "SCREEN","pe0,::pa 520,920;::pe1,::tx Revision Date:      "
237: wrt "SCREEN",D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6];wtb "SCREEN",3,10
238: for L=1 to 7 by 2;if I$[L,L]="0";" "}I$[L,L]
239: next L;I$[1,2]&"d "&I$[3,4]&"h "&I$[5,6]&"m "&I$[7,8]&"s")R$
240: wtb "SCREEN","pe0,::pa 520,890;::pe1,::tx Measurement Interval:"
241: wrt "SCREEN",R$;wtb "SCREEN",3,10
242: wtb "SCREEN","pe0,::pa 520,860;::pe1,::tx Counter Gate Time:   "
243: wrt "SCREEN",str(G)&" msec";wtb "SCREEN",3,10
244: wtb "SCREEN","pe0,::pa 32,770;::pe1,::"
245: wtb "SCREEN","tx   ####  Temperature Control  ####",3,10
246: wtb "SCREEN","pe0,::pa 553,770;::pe1,::"
247: wtb "SCREEN","tx   ####  Scanner Control  ####",3,10
248: wtb "SCREEN","pe0,::pa 25,710;::pe1,::"
249: wtb "SCREEN","tx      Temperature      Duration",3,10
250: wtb "SCREEN","pe0,::pa 25,685;::pe1,::"
251: wtb "SCREEN","tx ##    (deg C)     Days Hrs Mins Secs",3,10
252: wtb "SCREEN","pe0,::pa 25,665;::pe1,::"
253: wtb "SCREEN","tx ##   ##########   #### ### #### ####",3,10
254: for L=1 to A;str(L))R$;R$[2])R$;if L<10;" "&R$)R$
255: wtb "SCREEN","pe0,::pa 25,"&str(660-25L)&";::pe1,::tx "&R$,3,10
256: str(A[L]))R$;if abs(A[L])<10;" "&R$)R$
257: wtb "SCREEN","pe0,::pa 127,"&str(660-25L)&";::pe1,::tx "&R$,3,10
258: A$[L])R$;R$[1,2])R$;if R$[1,1]="0";" ")R$[1,1]
259: wtb "SCREEN","pe0,::pa 264,"&str(660-25L)&";::pe1,::tx "&R$,3,10
260: A$[L])R$;R$[3,4])R$;if R$[1,1]="0";" ")R$[1,1]
261: wtb "SCREEN","pe0,::pa 320,"&str(660-25L)&";::pe1,::tx "&R$,3,10
262: A$[L])R$;R$[5,6])R$;if R$[1,1]="0";" ")R$[1,1]
263: wtb "SCREEN","pe0,::pa 372,"&str(660-25L)&";::pe1,::tx "&R$,3,10
264: wtb "SCREEN","pe0,::pa 444,"&str(660-25L)&";::pe1,::tx 0",3,10;next L
```

```
267: wtb "SCREEN",*pe0,,,pe 530,685,,,pe1,,,"
268: wtb "SCREEN",*tx ##     Number      Name       Function",3,10
269: wtb "SCREEN",*pe0,,,pe 530,665,,,pe1,,,"
270: wtb "SCREEN",*tx ##     ######     ######     ########",3,10
271: for L=1 to 8;str(L)R$;R$[2])R$;if L<10;" "&R$)R$
272: wtb "SCREEN",*pe0,,,pe 531,"&str(660-25L)&";,,,pe1,,,tx "&R$,3,10
273: str(B[L]))R$;if B[L]<10;" "&R$)R$
274: wtb "SCREEN",*pe0,,,pe 613,"&str(660-25L)&";,,,pe1,,,tx "&R$,3,10
275: B$[L])R$;for M=1 to int(5-len(R$)/2);" "&R$)R$;next M
276: wtb "SCREEN",*pe0,,,pe 700,"&str(660-25L)&";,,,pe1,,,tx "&R$,3,10
277: if C$[L]="D";"DC V")R$
278: if C$[L]="A";"AC V")R$
279: if C$[L]="O";"Ohms")R$
280: wtb "SCREEN",*pe0,,,pe 879,"&str(660-25L)&";,,,pe1,,,tx "&R$,3,10;next L
281: gto "START"
282:
283: "Printer":
284: psc 705;hdcpy 1;pclr;scl 0,1023,0,1023
285: plt 0,0,1;plt 1023,0,2;plt 1023,1023;plt 0,1023;plt 0,0
286: xax 767;xax 818;xax 972;yax 512,0,0,818;fxd 0
287: plt 317,989,1;lbl "   Thermal Vacuum Test Facility"
288: plt 30,933,1;lbl "Program: INIT"
289: plt 30,903,1;lbl "Written: 12-17-88"
290: plt 30,873,1;lbl "For:     Space Applications Branch"
291: plt 30,843,1;lbl "         US Naval Research Lab"
292: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])R$
293: plt 540,933,1;lbl "Revision Date:        "&R$
294: plt 540,903,1;lbl "Measurement Interval:"
295: for L=1 to 7 by 2;if I$[L,L]="0";" ")I$[L,L]
296: next L;lbl I$[1,2]&"d "&I$[3,4]&"h "&I$[5,6]&"m "&I$[7,8]&"s"
297: plt 540,873,1;lbl "Counter Gate Time:   "&str(G)&" msec"
298: plt 72,785,1;lbl "####  Temperature Control  ####"
299: plt 610,785,1;lbl "####  Scanner Control  ####"
300: plt 35,710,1;lbl "     Temperature        Duration"
301: plt 35,685,1;lbl "##     (deg C)     Days Hrs Mins Secs"
302: plt 35,665,1;lbl "##   ############   #### ### #### ####"
303: for L=1 to A;str(L)R$;R$[2])R$;if L<10;" "&R$)R$
304: plt 31,660-25L,1;lbl R$
305: str(A[L]))R$;if abs(A[L])<10;" "&R$)R$
306: plt 132,660-25L,1;lbl R$
307: A$[L])R$;R$[1,2])R$;if R$[1,1]="0";" ")R$[1,1]
308: plt 266,660-25L,1;lbl R$
309: A$[L])R$;R$[3,4])R$;if R$[1,1]="0";" ")R$[1,1]
310: plt 319,660-25L,1;lbl R$
311: A$[L])R$;R$[5,6])R$;if R$[1,1]="0";" ")R$[1,1]
312: plt 371,660-25L,1;lbl R$
313: plt 442,660-25L,1;lbl "0";next L
314: plt 556,710,1;lbl "     Channel   Channel      Channel"
315: plt 550,685,1;lbl "##    Number     Name       Function"
316: plt 550,665,1;lbl "##    ######    ######     ########"
317: for L=1 to 8;str(L)R$;R$[2])R$;if L<10;" "&R$)R$
318: plt 551,660-25L,1;lbl R$
319: str(B[L]))R$;if B[L]<10;" "&R$)R$
320: plt 629,660-25L,1;lbl R$
321: B$[L])R$;for M=1 to int(5-len(R$)/2);" "&R$)R$;next M
322: plt 714,660-25L,1;lbl R$
323: if C$[L]="D";"DC V")R$
324: if C$[L]="A";"AC V")R$
325: if C$[L]="O";"Ohms")R$
326: plt 886,660-25L,1;lbl R$;next L;wtb "PRINTER",12,13,27,69;gto "START"
327: end
#2862
```

9.2 Program: GATHER

```
0: "PROGRAM:  THERMAL VACUUM TEST FACILITY MAIN PROGRAM";
1:
2: "*************************************************";
3: "*  Kevin Miller              *  File: GATHER      *";
4: "*  January 14, 1988          *  Update: July 6, 1983 *";
5: "*  Naval Research Laboratory *  Output Device: Multiple *";
6: "*  Space Applications Branch *  Select Codes: 7,8,9  *";
7: "*************************************************";
8:
9: "DECLARE VARIABLES AND DEVICES";
10: dsp "Insert Softcopy Graphics Tape";stp
11: idb 1;dev "KEPCO",701,"BCD",707,"SCANNER",709,"PRINTER",706,"PLOTTER",705
12: dev "COUNTER",712,"VOLTMETER",722,"SCREEN",718,"CLOCK",9
13: "VARIABLES IN THIS PROGRAM";dim C[25],T[4],K[25],I$[7],D$[6],K$[25,6]
14: dim Q$[16],C$[25,10],J$[25],E$[14],Y$[2],G$[1],T$[14],F$[6]
15: dim I,Q,G,N,K,P,C,A,J,B,D
16: "VARIABLES IN CHAINED PROGRAMS";dim H[2],V[2],M[25],S[2]
17: dim R$[15],U$[7],M$[40]
18: dim R,D,M,L,S,E,U,V,H,W,X,Y,Z
19: drive 0;files #:1,INIT.V:0,INIT.D:0;getk "KEYS"
20:
21: "READ INITIALIZATION DATA FROM INIT.V OR INIT.D";
22: ent "Variable or Default Init File",T$;2)T[4];if cap(T$[1,1])="D";3)T[4]
23: rread T[4],1,D$,T[1],J,T$,G;1)T[3])K
24: "TEMPERATURES";rread T[4],2,K[1],K$[1]
25: for B=2 to T[1];sread T[4],K[B],K$[B];next B
26: "CHANNELS";if J=0;jmp 3
27: rread T[4],4,C[1],C$[1],J$[1]
28: for B=2 to J;sread T[4],C[B],C$[B],J$[B];next B
29: 86400val(T$[1,2])+3600val(T$[3,4])+60val(T$[5,6])+val(T$[7,8])}I
30: "2187>-<;;"}T$;T$[6-log(G),6-log(G)]}G$
31:
32: "STARTUP DIALOG";
33: wrt "CLOCK","R";red "CLOCK",T$;spc 3
34: prt "***************","      GATHER","***************";spc 2
35: prt "Today's Date:"," "&T$[1,2]&"-"&T$[4,5]&"  "&T$[7];spc 2
36: "START";wrt "CLOCK","R";ent "Start Test:   MMDDHHMMSS",T$
37: if flg13;red "CLOCK",T$;T$[1,2]&T$[4,5]&T$[7,8]&T$[10,11]&T$[13,14]}T$
38: if len(T$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
39: T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10]}M$
40: prt "Start Test:"," "&M$;spc 2;T$)E$;"TIME")Q)T[2]
41: ent "End Test:  MMDDHHMMSS",E$
42: if flg13;dsp "An End Date MUST be Specified.";wait 1500;jmp -1
43: if not len(E$)=10;dsp "Not in Proper Form";wait 1500;jmp -2
44: E$[1,2]&"-"&E$[3,4]&"  "&E$[5,6]&":"&E$[7,8]&":"&E$[9,10]}M$
45: prt "End Test:"," "&M$;spc 2;if sgn("TIME"-Q)Q)-1;jmp 3
46: dsp "Test Must BEGIN Before It ENDS.";wait 1500
47: dsp "Try Again.";wait 1000;gto "START"
48: ent "Current Year:   YY",Y$
49: if val(Y$)<80;dsp "Too Small";wait 1500;jmp -1
50: if val(Y$)>90;dsp "Too Large";wait 1500;jmp -2
51: if rds(1)>120;dsp "Insert POWER FAIL TAPE";stp ;jmp 0
52: int(Q/I)}N;irt 0,K[1]}Q;ref 1,G$,T$,F$,I,B
53: if N=0;dsp "Time Interval Too Small";wait 1500;gto "START"
54: "FILE";ent "File Where Data is to be Stored",F$
55: drive 1;assgn F$,1,1,Q;if Q=1;gto "OPEN"
56: dsp F$," has Already Been Opened";wait 1500;gto "FILE"
57: "OPEN";open F$,N+4;assgn F$,1;T$)E$;prt "File: "&F$;spc 3
58: "WAIT";wrt "CLOCK","R";red "CLOCK",T$
59: if T$[1,2]&T$[4,5]&T$[7,8]&T$[10,11]&T$[13,14]<E$;jmp -1
60:
61: "READ INITIAL DEVICE INFORMATION";
62: wrt "CLOCK","R";red "CLOCK",T$;T$[1,2]&T$[4,5]&Y$)T$
63: wrt "COUNTER","F2G-BC0A+AB+ALA0.00LB0.00H0D1E1N0J0";red "COUNTER",P
64: wrt "COUNTER","F5G-GC0A+AB+ALA0.00LB0.00H0D1E1N0J0";red "COUNTER",C
65: rprt 1,1,G,N,I,T$,P,J,"end";for B=1 to J;sprt 1,C[B],C$[B],J$[B],B];next B
66: 999K[1]/105)T,"0123456789ABCDEF")M$;"+")T$;if T<0;"-")T$;abs(T)}T
67: for B=2 to 0 by -1;int(T/16^B)}C;M$[C+1,C+1]}T$[4-B,4-B];T-C16^B}T;next B
68: wrt "BCD",T$;if T[1]>1;"00"&K$[1]&"00")E$;"TIME"+T[2])T[2]
69:
70: "SET CLOCK TO INTERRUPT";
71: ami "CLOCK","TAKE READINGS AND STORE";eir "CLOCK"
72: fmt 1,"U1H,U1=01,U1P",f10.0,"/U1G";wrt "CLOCK.1",10001
73: wrt "CLOCK","U2H,U2=02,U2P5000/U2G"
74:
75: "NEXT PROGRAM";
76: for A=0 to 11;if flgA;cfg A;fxd 0;gsb +7
77: next A;len(I$))A;if I$[1,1]=" ";fxd 0;str(N-K+1))I$;I$[2])I$;gto +2
78: " ")I$[1,1];if A>1;" ")I$[2,2];if A>2;" ")I$[3,3];if A>3;" ")I$[4,4]
79: dsp "Select Program Key          ["&I$&"]";wait 300;lbl
80: if K<=N;gto "NEXT PROGRAM"
81: dsp "Execution Complete";wrt "COUNTER","12E0"
82: wrt "KEPCO","000000";drive 0;stp
83: drive 0;jmp A+1
84: "0";chain ".INI18",144,144
85: "1";chain ".INT18",144,144
86: "2";chain ".OFF18",144,144
```

```
87: "3";chain ".SIG10",144,144
88: "4";chain ".PLD10",144,144
89: "5";chain ".QUI16",144,144
90: "6";chain ".INI05",144,144
91: "7";chain ".INT05",144,144
92: "8";chain ".OFF05",144,144
93: "9";chain ".SIG05",144,144
94: "10";chain ".PLD05",144,144
95: "11";chain ".SCA06",144,144
96:
97: "TAKE READINGS AND STORE";
98: wrt "CLOCK","T";if not bit(0,rdb("CLOCK"));gto "TED AND CHAMBER CONTROL"
99: wrt "CLOCK","R";red "CLOCK",T0
100: T0[1,2]&T0[4,5]&T0[7,8]&T0[10,11]&T0[13,14])T0;T0)E0;fmt e13.6
101: if K-1;K[1])Q;ref 1,G0,T0,F0,I,0
102: wrt "COUNTER","F5G-6C0A-AB-ALA0.00LB0.00M0D1E1N0J0";red "COUNTER",C
103: drive 1;rprt 1,K-4,T0,C,"end";for B-1 to J
104: if J0[B,B]-"D";wrt "VOLTMETER","F1R7M3A1N0D0T3"
105: if J0[B,B]-"A";wrt "VOLTMETER","F2R7M3A1N0D0T3"
106: if J0[B,B]-"O";wrt "VOLTMETER","F4R7M3A1N0D0T3"
107: fmt 2,"C",fz2.0,"E";wrt "SCANNER.2",C[B]
108: 999K(T[3])/105)T;"0123456789ABCDEF")M0;"-")T0;if T<0;"-")T0;abs(T))T
109: for 0-2 to 0 by -1;int(T/16^0))C;M0[C+1,C+1])T0[4-0,4-0];T-C16^0)T;next 0
110: fmt 2,c,"FFFFFF",4c;wrt "BCD.2",M0[16-C[B]mod10)C,C],T0;wait 100
111: trg "VOLTMETER";wait 200;red "VOLTMETER",C;sprt 1,C,"end"
112: next B;if T[3]-T[1];gto -6
113: "0123456789ABCDEF")M0;if "TIME"<T[2]-.1;gto +5
114: 999K(T[3]+1)T[3])/105)T;"-")T0;if T<0;"-")T0;abs(T))T
115: for B-2 to 0 by -1;int(T/16^B))C,M0[C+1,C+1])T0[4-B,4-B];T-C16^B)T;next B
116: wrt "BCD",T0;if T[3]<T[1];"00"&K0[(T[3])&"00")E0;"TIME"+T[2])T[2]
117: idf 1,G0,T0,F0,1,0;K(T[3]))Q;ref 1,G0,T0,F0,I,0
118: drive 0;assgn "MEMORY",3,0,0;if Q#1;till "MEMORY"
119: savem "MEMORY";drive 1;K+1)K;eir "CLOCK";iret
120:
121: "TED AND CHAMBER CONTROL";
122: 999K(T[3])/105)Q;"0123456789ABCDEF")Q0;"-")T0;if Q<0;"-")T0;abs(Q))Q
123: for B-2 to 0 by -1;int(Q/16^B))C;Q0[C+1,C+1])T0[4-B,4-B];Q-C16^B)Q;next B
124: wrt "BCD","EFFFFFF"&T0;wrt "SCANNER","CJ1E";trg "VOLTMETER"
125: wait 200;red "VOLTMETER",D;1000)D
126: wrt "BCD","FFFFFFF"&T0;wrt "SCANNER","CJ0E";trg "VOLTMETER"
127: wait 200;red "VOLTMETER",C;100C)C;if K(T[3])-C-0;(C-D)/15)Q;gto +2
128: 5.6(K(T[3])-D)+sgn(K(T[3])-C)181.7*10^(.431og(abs(K(T[3])-C))))Q
129: fxd 0;if abs(Q)>400;"400")Q0;gto +2
130: str(int(Q)))Q0;Q0[2])Q0;if abs(Q)<100;"0"&Q0)Q0;if abs(Q)<10;"0"&Q0)Q0
131: "4"&Q0)Q0;if Q<0;"5")Q0[1,1]
132: "60")Q0[5];if Q>400;"400")Q0[2,4]
133: wrt "KEPCO",Q0;eir "CLOCK";iret
134:
135: "TIME";
136: 0)p1;for D-1 to val(E0[1,2])-1
137: if D-1 or D-3 or D-5 or D-7 or D-8 or D-10 or D-12;p1+2678400)p1;jmp 3
138: if D-4 or D-6 or D-9 or D-11;p1+2592000)p1;jmp 2
139: if D-2;p1+2419200)p1
140: next B
141: p1+86400val(E0[3,4])+3600val(E0[5,6]))p1
142: ret p1+60val(E0[7,8])+val(E0[9,10])
143:
144: end
*10545
```

9.3 Program: REBOOT

```
0: "PROGRAM: POWER FAILURE BOOTSTRAP";
1:
2: "*************************************************************";
3: "*  Kevin Miller                  *  File: REBOOT              *";
4: "*  January 14, 1980              *  Update: July 20, 1983     *";
5: "*  Naval Research Laboratory     *  Output Devices: 5345, 3455 *";
6: "*  Space Applications Branch     *  Select Codes: 712, 722, 9  *";
7: "*************************************************************";
8:
9: dev "COUNTER",712,"VOLTMETER",722,"CLOCK",9
10: dim G$[1],T$[14],F$[6],I,Q,A,C,T,U
11: idf 1,G$,T$,F$,I,Q
12: drive 1;assgn F$,1
13: wrt "VOLTMETER","TJ"
14: wrt "COUNTER","F5G-6C0A+AB+ALA0.00LB0.00H0D1E1H0J0";red "COUNTER",C
15: "TIME"}T
16: wrt "CLOCK","R";red "CLOCK",T$
17: prt "Power fail","restart at:"," ",T$;spc 3
18: T$[1,2]&T$[4,5]&T$[7,8]&T$[10,11]&T$[13,14]}T$
19: "TIME"}U
20: if U>T;T+1}T;jmp 8
21: wrt "CLOCK","R";red "CLOCK",T$
22: T$[1,2]&T$[4,5]&T$[7,8]&T$[10,11]&T$[13,14]}T$
23: fxd 0;dsp "Seconds till restart:"&str(T-"TIME")
24: if T>"TIME";gto -3
25: fmt 1,"U1H,U1-01,U1P",f10.0,"/U1G"
26: wrt "CLOCK,1",1000
27: wrt "CLOCK","U2H,U2-02,U2P5000/U20"
28: drive 0;getm "MEMORY"
29: end
30: "TIME";
31: 0}p1;for A=1 to val(T$[1,2])-1
32: if A=1 or A=3 or A=5 or A=7 or A=8 or A=10 or A=12;p1+2678400}p1;jmp 3
33: if A=4 or A=6 or A=9 or A=11;p1+2592000}p1;jmp 2
34: if A=2;p1+2419200}p1
35: next A
36: p1+86400val(T$[3,4])+3600val(T$[5,6])}p1
37: ret p1+60val(T$[7,8])+val(T$[9,10])
#20447
```

9.4 Program: PATCH

```
0: "PROGRAM:  PATCH HALF-FILLED DATA FILES"
1:
2: "*************************************************************";
3: "*  Kevin Miller                 *  File: PATCH                 *";
4: "*  December 28, 1980            *  Update: December 28, 1980   *";
5: "*  Naval Research Laboratory    *  Output Device: 9885         *";
6: "*  Space Applications Branch    *  Select Code: 8              *";
7: "*************************************************************";
8:
9: "GET FILE NAME AND PATCH"
10: dim G,M,I,D$[6],T$[14],P,J,C[25],C$[25,10],J$[25],F$[6],Q,K,L,C
11: files #;1
12: ""}F$;ent "File to be Patched",F$;if flg13;jmp 0
13: drive 1;assgn F$,1,1,Q;if Q#1;jmp 2
14: dsp F$&" does not exist";wait 1500,;jmp -2
15: if Q#2 and Q#3 and Q#5 and Q#6;jmp 2
16: dsp F$&" is not a Data File";wait 1500,;jmp -4
17: assgn F$,1;0)K;on end 1,"Fix"
18: rread 1,1,G,M,I,D$,P,J
19: for L=1 to J;sread 1,C[L],C$[L],J$[L,L];next L
20: K+1)K;rread 1,K+4,T$,C;jmp 0
21: dsp "####  PROGRAM ERROR  ####";stp
22: "Fix":rprt 1,1,G,K-1)K,I,D$,P,J
23: for L=1 to J;sprt 1,C[L],C$[L],J$[L,L];next L
24: fxd 0;dsp "Old M ="&str(M)&"   New M ="&str(K);drive 8
25: end
#10412
```

9.5 Program: .INI05

```
0: "SUBPROGRAM:  INITIALIZATION FILE LIST";
1:
2: "*************************************************************";
3: "*  Kevin Miller                 *  File: .INI05               *";
4: "*  December 20, 1980            *  Update: December 20, 1980  *";
5: "*  Naval Research Laboratory    *  Output Device: 7245        *";
6: "*  Space Applications Branch    *  Select Code: 705           *";
7: "*************************************************************";
8:
9: "DISPLAY ON PRINTER";
10: rread T[4],1,D$,T[1],J,M$,0
11: hdcpy 1;psc 705;pclr;scl 0,1023,0,1023
12: plt 0,0,1;plt 1023,0,2;plt 1023,1023;plt 0,1023;plt 0,0
13: xax 767;xax 818;xax 972;yax 512,0,0,818;fxd 0
14: plt 317,989,1;lbl "Precision Oscillator Test Facility"
15: plt 30,933,1;lbl "Program: INIT"
16: plt 30,903,1;lbl "Written: 12-21-80"
17: plt 30,873,1;lbl "For:     Space Applications Branch"
18: plt 30,843,1;lbl "         US Naval Research Lab"
19: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])R$
20: plt 540,933,1;lbl "Revision Date:        "&R$
21: plt 540,903,1;lbl "Measurement Interval:"
22: for L=1 to 7 by 2;if M$[L,L]="0";" ")M$[L,L]
23: next L;lbl M$[1,2]&"d "&M$[3,4]&"h "&M$[5,6]&"m "&M$[7,8]&"s"
24: fxd 0;plt 540,873,1;lbl "Counter Gate Time:   "&str(G)&" msec"
25: plt 72,785,1;lbl "#### Temperature Control ####"
26: plt 610,785,1;lbl "#### Scanner Control ####"
27: plt 35,710,1;lbl "      Temperature          Duration"
28: plt 35,685,1;lbl "##      (deg C)      Days Hrs Mins Secs"
29: plt 35,665,1;lbl "##   ############    #### ### #### ####"
30: for L=1 to T[1];fxd 0;str(L)R$;R$[2]R$;if L<10;" "&R$)R$
31: plt 31,660-25L,1;lbl R$
32: fxd 0;str(K[L])R$;if abs(K[L])<10;" "&R$)R$
33: plt 132,660-25L,1;lbl R$
34: K$[L])R$;R$[1,2])R$;if R$[1,1]="0";" ")R$[1,1]
35: plt 266,660-25L,1;lbl R$
36: K$[L])R$;R$[3,4])R$;if R$[1,1]="0";" ")R$[1,1]
37: plt 319,660-25L,1;lbl R$
38: K$[L])R$;R$[5,6])R$;if R$[1,1]="0";" ")R$[1,1]
39: plt 371,660-25L,1;lbl R$
40: plt 442,660-25L,1;lbl "0";next L
41: plt 556,710,1;lbl "     Channel    Channel      Channel"
42: plt 550,685,1;lbl "##    Number      Name      Function"
43: plt 550,665,1;lbl "##    ######    ######    ########"
44: for L=1 to J;fxd 0;str(L)R$;R$[2]R$;if L<10;" "&R$)R$
45: plt 551,660-25L,1;lbl R$
46: fxd 0;str(C[L])R$;if C[L]<10;" "&R$)R$
47: plt 629,660-25L,1;lbl R$
48: C$[L])R$;for M=1 to int(5-len(R$)/2);" "&R$)R$;next M
49: plt 714,660-25L,1;lbl R$
50: if J$[L,L]="D";"DC V")R$
51: if J$[L,L]="A";"AC V")R$
52: if J$[L,L]="O";"Ohms")R$
53: plt 886,660-25L,1;lbl R$;next L
54: fxd 0;plt 540,843,1;lbl "Temperature Setpoint:"&str(T[3])
55: wtb "PRINTER",12,13,27,69
56: "RETURN";ret
≡00
```

9.6 Program: .INI18

```
0: "SUBPROGRAM:  INITIALIZATION FILE LIST";
1:
2: "*************************************************************";
3: "*  Kevin Miller                 *  File: .INI18                *";
4: "*  December 20, 1980            *  Update: December 20, 1980   *";
5: "*  Naval Research Laboratory    *  Output Device: 1350         *";
6: "*  Space Applications Branch    *  Select Code: 718            *";
7: "*************************************************************";
8:
9: "DISPLAY ON SCREEN";
10: rread T[4],1,D$,T[1],J,M$,6
11: fxd 0;hdcpy 0;psc 718;pclr
12: scl 0,2,0,20;xax 0;xax 15;xax 16;xax 19;xax 20
13: yax 0;yax 1,0,0,16;yax 2
14: wtb "SCREEN",3,"em::sn::sx::um::pe0,::pa287,967;:;"
15: wtb "SCREEN","pe1,::cs,::tx Precision Oscillator Test Facility",3,10
16: wtb "SCREEN","pe0,::pa 30,920;::pe1,::tx Program: INIT",3,10
17: wtb "SCREEN","pe0,::pa 30,890;::pe1,::tx Written: 12-20-80",3,10
18: wtb "SCREEN","pe0,::pa 30,860;::pe1,:;"
19: wtb "SCREEN","tx For:     Space Applications Branch",3,10
20: wtb "SCREEN","pe0,::pa 30,830;::pe1,:;"
21: wtb "SCREEN","tx          US Naval Research Lab",3,10
22: wtb "SCREEN","pe0,::pa 520,920;::pe1,::tx Revision Date:        "
23: wrt "SCREEN",D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6];wtb "SCREEN",3,10
24: for L=1 to 7 by 2;if M$[L,L]="0";" "}M$[L,L]
25: next L;M$[1,2]&"d "&M$[3,4]&"h "&M$[5,6]&"m "&M$[7,8]&"s"}R$
26: wtb "SCREEN","pe0,::pa 520,890;::pe1,::tx Measurement Interval:"
27: wrt "SCREEN",R$;wtb "SCREEN",3,10
28: wtb "SCREEN","pe0,::pa 520,860;::pe1,::tx Counter Gate Time:    "
29: fxd 0;wrt "SCREEN",str(G)&" msec";wtb "SCREEN",3,10
30: wtb "SCREEN","pe0,::pa 32,770;::pe1,:;"
31: wtb "SCREEN","tx   ####  Temperature Control  ####",3,10
32: wtb "SCREEN","pe0,::pa 553,770;::pe1,:;"
33: wtb "SCREEN","tx   ####  Scanner Control  ####",3,10
34: wtb "SCREEN","pe0,::pa 25,710;::pe1,:;"
35: wtb "SCREEN","tx      Temperature          Duration",3,10
36: wtb "SCREEN","pe0,::pa 25,685;::pe1,:;"
37: wtb "SCREEN","tx ##      (deg C)      Days Hrs Mins Secs",3,10
38: wtb "SCREEN","pe0,::pa 25,665;::pe1,:;"
39: wtb "SCREEN","tx ##    ##########    #### ### #### ####",3,10
40: fxd 0;for L=1 to T[1];str(L))R$;R$[2])R$;if L<10;" "&R$)R$
41: fxd 0;wtb "SCREEN","pe0,::pa 25,"&str(660-25L)&";::pe1,::tx "&R$,3,10
42: fxd 0;str(K[L]))R$;if abs(K[L])<10;" "&R$)R$
43: fxd 0;wtb "SCREEN","pe0,::pa 127,"&str(660-25L)&";::pe1,::tx "&R$,3,10
44: K$[L])R$;R$[1,2])R$;if R$[1,1]="0";" ")R$[1,1]
45: fxd 0;wtb "SCREEN","pe0,::pa 264,"&str(660-25L)&";::pe1,::tx "&R$,3,10
46: K$[L])R$;R$[3,4])R$;if R$[1,1]="0";" ")R$[1,1]
47: fxd 0;wtb "SCREEN","pe0,::pa 320,"&str(660-25L)&";::pe1,::tx "&R$,3,10
48: K$[L])R$;R$[5,6])R$;if R$[1,1]="0";" ")R$[1,1]
49: fxd 0;wtb "SCREEN","pe0,::pa 372,"&str(660-25L)&";::pe1,::tx "&R$,3,10
50: fxd 0;wtb "SCREEN","pe0,::pa 444,"&str(660-25L)&";::pe1,::tx 0",3,10
51: next L;wtb "SCREEN","pe0,::pa 536,710;::pe1,:;"
52: wtb "SCREEN","tx        Channel    Channel      Channel",3,10
53: wtb "SCREEN","pe0,::pa 530,685;::pe1,:;"
54: wtb "SCREEN","tx ##      Number      Name       Function",3,10
55: wtb "SCREEN","pe0,::pa 530,665;::pe1,:;"
56: wtb "SCREEN","tx ##      ######     ######     ########",3,10
57: for L=1 to J;fxd 0;str(L))R$;R$[2])R$;if L<10;" "&R$)R$
58: fxd 0;wtb "SCREEN","pe0,::pa 531,"&str(660-25L)&";::pe1,::tx "&R$,3,10
59: fxd 0;str(C[L]))R$;if C[L]<10;" "&R$)R$
60: fxd 0;wtb "SCREEN","pe0,::pa 613,"&str(660-25L)&";::pe1,::tx "&R$,3,10
61: C$[L])R$;for M=1 to int(5-len(R$)/2);" "&R$)R$;next M
62: fxd 0;wtb "SCREEN","pe0,::pa 700,"&str(660-25L)&";::pe1,::tx "&R$,3,10
63: if J$[L,L]="D";"DC V")R$
64: if J$[L,L]="A";"AC V")R$
65: if J$[L,L]="O";"Ohms")R$
66: fxd 0;wtb "SCREEN","pe0,::pa 879,"&str(660-25L)&";::pe1,::tx "&R$,3,10
67: next L;mfile 1;wtb "SCREEN","pe0,::pa 520,830;::pe1,:;"
68: fxd 0;wtb "SCREEN","tx Temperature Setpoint:"&str(T[3]),3,10;flsh 1
69: "RETURN";ret
*3160
```

9.7 Program: INT05

```
0: "PROGRAM:  TIME INTERVAL PLOT"
1:
2: "*************************************************"
3: "*  Kevin Miller                 *  File: INT05     *"
4: "*  January 14, 1980             *  Update: December 20, 1980  *"
5: "*  Naval Research Laboratory    *  Output Device: 7245        *"
6: "*  Space Applications Branch    *  Select Code: 705           *"
7: "*************************************************"
8:
9: "GET FILE NAME AND PLOT INTERVAL"
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PRINTER",706,"PLOTTER",705,"CLOCK",9;flt 11
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],V[2],B$[12],E$[12]
13: dim C,D,F,G,I,K,L,M,N,P,Q,4]K
14: ""]F$;ent "Enter Data File Name",F$;if flg13;jmp 0
15: asgn F$,1,1,B
16: if Q=1;dsp F$&" does not exist";wait 1500;;jmp -2
17: asgn F$,1,1;rread 1,1,G,N,I,D$,P
18: if N<3;dsp "Not Enough Data for Plot";stp
19: ent "Plot Data from:  MMDDHHMMSS",B$
20: if flg13,"0000000000"]B$
21: if len(B$)≠10;dsp "Not in Proper Form";wait 1500;;jmp -2
22: ent "Plot Data to:  MMDDHHMMSS",E$
23: if flg13,"9999999999"]E$
24: if len(E$)≠10;dsp "Not in Proper Form";wait 1500;;jmp -2
25:
26: "CALCULATE UNITS FOR AXES"
27: rread 1,K+1]K,T$,C;if T$<B$;jmp 0
28: "TIME"]H[1];C]V[1]]V[2],1]F
29: for L=K+1 to N
30: rread 1,L,T$,C;if T$>E$;;jmp 4
31: if C<V[1];C]V[1]
32: if C>V[2];C]V[2]
33: F+1]F;next L
34: rread 1,L-1,T$;"TIME"]H[2]
35: if H[2]-H[1]>86399;"DAYS"]U$;86400]D;;jmp 4
36: if H[2]-H[1]>3599;"HOURS"]U$;3600]D;;jmp 3
37: if H[2]-H[1]>59;"MINUTES"]U$;60]D;;jmp 2
38: "SECONDS"]U$;1]D
39: Dint(H[1]/D)]H[1],Dint(H[2]/D+1)]H[2]
40: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6]]T$;4]K
41:
42: "INITIALIZE PLOT AREA"
43: fxd 0;hdcpy 1;pse 705;scl -7.125,0,0,10.5;plt 0,8,1
44: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
45: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
46: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
47: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.55,1;lbl "Time Interval Plot"
48: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
49: plt -1/16,3.52,1;lbl F$
50: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
51: plt -1/16,5.1,1;lbl str(F)
52: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
53: fxd 2;lbl str(1/P*1e-6)&" MHz"
54: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
55: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
56: plt -9/16,9.25,1;lbl "Counter Gate";plt -5.5/16,9.6,1;lbl "Time"
57: plt -1/16,9.35,1;fxd 0;lbl str(G)&" msec";plt -1.375,10.4,1
58: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4;flt 9
59: scl V[2],1.23913V[1]-.23913V[2],H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
60:
61: "LABEL AXES"
62: for L=V[1]+((V[2]-V[1])/10)C) to V[2]-C by C;plt L,H[2],1
63: plt L,H[2]-(H[2]-H[1])/100,2;next L
64: for L=H[2]-D to H[1]+D by -D;plt V[2],L,1;plt V[2]-C/7,L,2;next L
65: csiz 1,2,1,90;for L=V[2] to V[1] by -C;plt L-C/10,H[1]-(H[2]-H[1])/17,1
66: if 1e9L>100;fxd 2;lbl str(L*1e9);gto +3
67: if 1e9L>10;fxd 2;lbl " "&str(L*1e9);gto +2
68: fxd 2;lbl "  "&str(L*1e9)
69: if L≠V[1] or L≠V[2];plt L,H[1],1;plt L,H[1]+(H[2]-H[1])/100,2
70: next L;csiz 1.5,2,1,180;plt V[1]+2.51C,H[1]-(H[2]-H[1])/12,1
71: lbl "Time Interval A to B in nanoseconds";csiz 1,2,1,90
72: for L=H[1] to H[2] by D;if L=V[1] or L=V[2];jmp 2
73: plt V[1]+C/7,L,1;plt V[1],L,2;plt V[1]-C/2.4,L-(H[2]-H[1])/100,1
74: L)Q;for M=1 to 12
75: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);;jmp 3
76: if Q-2764800<0;gto "PRINT"
77: Q-2678400)Q
78: if not (M=4 or M=6 or M=9 or M=11);;jmp 3
79: if Q-2678400<0;gto "PRINT"
80: Q-2592000)Q
81: if not M=2;;jmp 3
82: if Q-2505600<0;gto "PRINT"
83: Q-2419200)Q
84: next M
85: "PRINT";if D=86400;fxd 0;lbl str(Q/D)
86: if D=3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
```

```
87: if D-60;fxd 0;lbl str((Q-3600:nt(Q/3600))/D)
88: if D-1;fxd 0;lbl str((Q-60:nt(Q/60))/D)
89: next L;cslz 1.5,2,1,90;plt V[1]-.81C,M[1]+(M[2]-M[1])/3,1
90: if B$="0000000000";jmp 2
91: B$[1,2]&"-"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M$;jmp 2
92: rread 1,5,T$;T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
93: lbl "Time in "&U$&" beginning  "&M$
94:
95: "PLOT DATA";
96: rread 1,K+1)K,T$,C;if T$<B$;jmp 0
97: plt C,"TIME",1;cplt -.33,-.15;lbl "o";cplt -.67,.15
98: for L=K+1 to N;rread 1,L,T$,C;if T$>E$;jmp 2
99: plt C,"TIME",2;next L
100: pen;cplt -.33,-.15;lbl "o";wtb "PRINTER",12,13,27,69
101: end
102:
103: "TIME";
104: 0)Q;for M=1 to val(T$[1,2])-1
105: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or .12;Q+267840)Q;jmp 3
106: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;jmp 2
107: if M=2;Q+2419200)Q
108: next M
109: Q+86400val(T$[3,4])+3600val(T$[5,6]))Q
110: ret Q+60val(T$[7,8])+val(T$[9,10])
*26804
```

9.8 Program: INT18

```
0: "PROGRAM:  TIME INTERVAL PLOT"
1:
2: "*************************************************"
3: "*  Kevin Miller                 *  File: INT18     *"
4: "*  January 14, 1980             *  Update: December 28, 1980  *"
5: "*  Naval Research Laboratory    *  Output Device: 1350        *"
6: "*  Space Applications Branch    *  Select Code: 718           *"
7: "*************************************************"
8:
9: "GET FILE NAME AND PLOT INTERVAL"
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "CLOCK",9;flt 11
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],V[2],B$[12],E$[12]
13: dim C,D,F,G,I,K,L,M,N,P,Q,4]K
14: ""]F$;ent "Enter Data File Name",F$;if flg13;jmp 0
15: assgn F$,1,1,0
16: if Q=1;dsp F$&" does not exist";wait 1500;jmp -2
17: assgn F$,1,1;rread 1,1,G,M,I,D$,P
18: if M<3;dsp "Not Enough Data for Plot";stp
19: ent "Plot Data from:  MMDDHHMMSS",B$
20: if flg13;"0000000000"]B$
21: if len(B$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
22: ent "Plot Data to:  MMDDHHMMSS",E$
23: if flg13;"9999999999"]E$
24: if len(E$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
25:
26: "CALCULATE UNITS FOR AXES"
27: rread 1,K+1)K,T$,C;if T$<B$;jmp 0
28: 'DATE']H[1],C)V[1])V[2],1)F
29: for L=K+1 to M
30: rread 1,L,T$,C;if T$>E$;jmp 4
31: if C<V[1];C)V[1]
32: if C>V[2];C)V[2]
33: F+1)F;next L
34: rread 1,L-1,T$,'DATE']H[2]
35: if H[2]-H[1]>86399;"DAYS"]U$;86400)D;jmp 4
36: if H[2]-H[1]>3599;"HOURS"]U$;3600)D;jmp 3
37: if H[2]-H[1]>59;"MINUTES"]U$;60)D;jmp 2
38: "SECONDS"]U$;1)D
39: Dint(H[1]/D)]H[1];Dint(H[2]/D+1)]H[2]
40: 1)W;if ((H[2]-H[1])/D)R)>30;2)W;if int(R/2)#R/2;H[2]+D)H[2]
41: wrt "CLOCK","R";rred "CLOCK",T$;4)K
42:
43: "INITIALIZE PLOT AREA"
44: fxd 0;hdcpy 0;psc 718;pclr;scl 0,10.5,0,7.125;plt 0,0,1
45: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
46: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
47: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today"
48: plt .21,1/11,1;lbl T$[1,2]&"-"&T$[4,5]&"-"&"80"
49: plt 1.85,7.5/16,1;lbl "Program";plt 1.6,1/11,1;lbl "T1 Int Plot"
50: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
51: plt 3.4,1/11,1;lbl F$
52: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl "Plotted"
53: plt 4.95,1/11,1;lbl str(F)&" "
54: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.15,1/11,1
55: fxd 2;lbl str(1/P*1e-6)&" MHz"
56: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
57: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
58: plt 9.45,9/16,1;lbl "Gate";plt 9.45,5.7/16,1;lbl "Time"
59: plt 9.1,1/11,1;fxd 0;lbl str(G)&" msec ";plt 10.4,1.375,1
60: plt 10.4,7.125,2;plt 1,7.125;plt 1,1.375;plt 10.4,1.375;flt 9
61: scl H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94,1.23913V[1]-.23913V[2],V[2]
62:
63: "LABEL AXES"
64: for L=V[1]+((V[2]-V[1])/10)C) to V[2]-C by C;plt H[2],L,1
65: plt H[2]-(H[2]-H[1])/100,L,2;next L
66: for L=H[2]-WD to H[1]+WD by -WD;plt L,V[2],1;plt L,V[2]-C/7,2;next L
67: csiz .7,2,1;for L=V[2] to V[1] by -C;plt H[1]-(H[2]-H[1])/12,L-C/10,1
68: if 1e9L>100;fxd 1;lbl str(L*1e9);gto +3
69: if 1e9L>10;fxd 1;lbl " "&str(L*1e9)&" ";gto +2
70: fxd 1;lbl "  "&str(L*1e9)
71: if L#V[1] and L#V[2];plt H[1],L,1;plt H[1]+(H[2]-H[1])/100,L,2
72: next L;csiz 1.5,2,1,90;plt H[1]-(H[2]-H[1])/12,V[1]+2.1C,1
73: lbl "Time Interval A to B in nanoseconds";csiz .7,2,1
74: for L=H[1] to H[2] by WD;if L=H[1] or L=H[2];jmp 2
75: plt L,V[1]+C/7,1;plt L,V[1],2
76: plt L-(H[2]-H[1])/50,V[1]-.35C,1;L)Q;for M=1 to 12
77: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
78: if Q-2764800<0;gto "PRINT"
79: Q-2678400)Q
80: if not (M=4 or M=6 or M=9 or M=11);jmp 3
81: if Q-2678400<0;gto "PRINT"
82: Q-2592000)Q
83: if not M=2;jmp 3
84: if Q-2505600<0;gto "PRINT"
85: Q-2419200)Q
86: next M
```

```
87: "PRINT";if D=86400;fxd 0;lbl str(Q/D)
88: if D=3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
89: if D=60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
90: if D=1;fxd 0;lbl str((Q-60int(Q/60))/D)
91: next L;cwin 1.5,2,1;plt H[1]+(H[2]-H[1])/4.2,V[1]-.85C,1
92: if B$="0000000000";jmp 2
93: B$[1,2]&"-"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M$;jmp 2
94: rread 1,5,T$;T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
95: lbl "Time in "&U$&" beginning  "&M$&" "
96:
97: "PLOT DATA":
98: rread 1,K+1)K,T$,C;if T$<B$;jmp 8
99: plt 'DATE',C,1;cplt -.33,-.15;lbl "a";cplt -.67,.15
100: for L=K+1 to N;rread 1,L,T$,C;if T$>E$;jmp 2
101: plt 'DATE',C,2;next L
102: pen;cplt -.33,-.15;lbl "a"
103: end
104:
105: "DATE":
106: 0)Q;for M=1 to val(T$[1,2])-1
107: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;jmp 3
108: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;jmp 2
109: if M=2;Q+2419200)Q
110: next M
111: Q+86400val(T$[3,4])+3600val(T$[5,6]))Q
112: ret Q+60val(T$[7,8])+val(T$[9,10])
#31808
```

9.9 Program: .INT05

```
0: "SUBPROGRAM:  TIME INTERVAL PLOT"
1:
2: "*************************************************"
3: "*  Kevin Miller              *  File: .INT05       *"
4: "*  August 18, 1980           *  Update: December 20, 1980  *"
5: "*  Naval Research Laboratory *  Output Device: 7245       *"
6: "*  Space Applications Branch *  Select Code: 705          *"
7: "*************************************************"
8:
9:
10: "CALCULATE UNITS FOR AXES"
11: if K>2;jmp 3
12: spc 3;prt "Not Enough Data","Yet to do  TIME"
13: prt "INTERVAL Plot.";spc 3;gto "RETURN"
14: rread 1,5,R$,R
15: "DATE")H[1],R)V[1])V[2]
16: for L=2 to K-1)Z
17: rread 1,L+4,R$,R
18: if R<V[1],R)V[1]
19: if R>V[2],R)V[2]
20: next L;"DATE")H[2]
21: if H[2]-H[1]>86399;"DAYS")U$;86400)D;jmp 4
22: if H[2]-H[1]>3599;"HOURS")U$;3600)D;jmp 3
23: if H[2]-H[1]>59;"MINUTES")U$;60)D;jmp 2
24: "SECONDS")U$;1)D
25: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2]
26: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])R$
27:
28: "INITIALIZE PLOT AREA"
29: fxd 0;hdcpy 1;psc 705;pclr;scl -7.125,0,0,10.5;plt 0,0,1
30: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
31: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
32: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl R$
33: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.55,1;lbl "Time Interval Plot"
34: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
35: plt -1/16,3.52,1;lbl F$
36: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
37: plt -1/16,5.1,1;fxd 0;lbl str(Z)
38: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
39: fxd 2;lbl str(1/P*1e-6)&" MHz"
40: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
41: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
42: plt -9/16,9.25,1;lbl "Counter Gate";plt -5.5/16,9.6,1;lbl "Time"
43: plt -1/16,9.35,1;fxd 0;lbl str(G)&" msec";plt -1.375,10.4,1
44: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4;flt 9
45: scl V[2],1.23913V[1]-.23913V[2],H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
46:
47: "LABEL AXES"
48: for L=V[1]+((V[2]-V[1])/10)R) to V[2]-R by R;plt L,H[2],1
49: plt L,H[2]-(H[2]-H[1])/180,2;next L
50: for L=H[2]-D to H[1]+D by -D;plt V[2],L,1;plt V[2]-R/7,L,2;next L
51: csiz 1.2,1,90;for L=V[2] to V[1] by -R;plt L-R/10,H[1]-(H[2]-H[1])/17,1
52: if 1e9L>100;fxd 2;lbl str(L*1e9);gto +3
53: if 1e9L>10;fxd 2;lbl " "&str(L*1e9);gto +2
54: fxd 2;lbl "  "&str(L*1e9)
55: if L<V[1] or L>V[2];plt L,H[1],1;plt L,H[1]+(H[2]-H[1])/100,2
56: next L;csiz 1.5,2,1,180;plt V[1]+2.5iR,H[1]-(H[2]-H[1])/12,1
57: lbl "Time Interval A to B in nanoseconds";csiz 1,2,1,90
58: for L=H[1] to H[2] by D;if L=V[1] or L=V[2];jmp 2
59: plt V[1]+R/7,L,1;plt V[1],L,2;plt V[1]-R/2.4,L-(H[2]-H[1])/100,1
60: L)X;for M=1 to 12
61: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
62: if X-2764800<0;gto "PRINT"
63: X-2678400)X
64: if not (M=4 or M=6 or M=9 or M=11);jmp 3
65: if X-2678400<0;gto "PRINT"
66: X-2592000)X
67: if not M=2;jmp 3
68: if X-2505600<0;gto "PRINT"
69: X-2419200)X
70: next M
71: "PRINT";if D=86400;fxd 0;lbl str(X/D)
72: if D=3600;fxd 0;lbl str((X-86400int(X/86400))/D)
73: if D=60;fxd 0;lbl str((X-3600int(X/3600))/D)
74: if D=1;fxd 0;lbl str((X-60int(X/60))/D)
75: next L;csiz 1.5,2,1,90;plt V[1]-.81R,H[1]+(H[2]-H[1])/3,1
76: rread 1,5,R$,R$[1,2]&"-"&R$[3,4]&"  "&R$[5,6]&":"&R$[7,8]&":"&R$[9,10])M$
77: "1 "Time in "&U$&" beginning  "&M$
78:
79:  PLOT DATA"
80: rread 1,5,R$,R
81: plt R,"DATE",1;cplt -.33,-.15;lbl "o";cplt -.67,.15
82: for L=2 to Z
83: rread I,L+4,R$,R
84: plt R,"DATE",2
85: next L
86: cplt -.33,-.15;lbl "o"
```

```
87: wtb "PRINTER",10,13,27,69
88: "RETURN":ret
89:
90: "DATE":
91: 0)X;for M=1 to val(R$[1,2])-1
92: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;X=2678400)X;)mp 3
93: if M=4 or M=6 or M=9 or M=11;X=2592000)X;)mp 2
94: if M=2;X=2419200)X
95: next M
96: X=86400val(R$[3,4])+3600val(R$[5,6]))X
97: ret X+60val(R$[7,8])+val(R$[9,10])
#17097
```

9.10 Program: .INT18

```
0: "SUBPROGRAM:  TIME INTERVAL PLOT"
1:
2: "**************************************************"
3: "*  Kevin Miller              *  File: .INT18       *"
4: "*  December 19, 1980         *  Update: December 20, 1980  *"
5: "*  Naval Research Laboratory *  Output Device: 1350  *"
6: "*  Space Applications Branch *  Select Code: 718    *"
7: "**************************************************"
8:
9: "CALCULATE UNITS FOR AXES"
10: hdcpy 0;psc 718;flt 11
11: 0)R)L)Z)D)M)H[1])H[2])V[1])V[2];if K)2;jmp 3
12: spc 3;prt "Not Enough Data","Yet to do  TIME","INTERVAL Plot.";spc 3
13: gto "RETURN"
14: rread 1,5,R$,R,'DATE')H[1];R)V[1])V[2]
15: for L=2 to K-1)Z;rread 1,L*4,R$,R
16: if R<V[1];R)V[1]
17: if R)V[2];R)V[2]
18: next L;'DATE')H[2]
19: if H[2]-H[1]>86399,"DAYS")U$;86400)D;jmp 4
20: if H[2]-H[1]>3599,"HOURS")U$;3600)D;jmp 3
21: if H[2]-H[1]>59,"MINUTES")U$;60)D;jmp 2
22: "SECONDS")U$;1)D
23: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2]
24: 1)W;if ((H[2]-H[1])/D)R)>30;2)W;if int(R/2)≠R/2;H[2]+D)H[2]
25:
26: "INITIALIZE PLOT AREA"
27: fxd 0;plr;sel 0,10.5,0,7.125;plt 0,0,1
28: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
29: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
30: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today"
31: plt .21,1/11,1;lbl D$[1,2]&"-"&D$[4,5]&"-"&D$[5,6]&" "
32: plt 1.85,7.5/16,1;lbl "Program";plt 1.6,1/11,1;lbl "T1 Int Plot"
33: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
34: plt 3.4,1/11,1;lbl F$&" "
35: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl "Plotted"
36: fxd 0;plt 4.95,1/11,1;lbl str(Z)&" "
37: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.15,1/11,1
38: fxd 2;lbl str(1/P*1e-6)&" MHz"
39: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
40: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
41: plt 9.45,9/16,1;lbl "Gate";plt 9.45,5.7/16,1;lbl "Time"
42: plt 9.1,1/11,1;fxd 0;lbl str(G)&" msec ";plt 10.4,1.375,1
43: plt 10.4,7.125,2;plt 1,7.125;plt 1,1.375;plt 10.4,1.375;flt 9
44: sel H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94,1.23913V[1]-.23913V[2],V[2]
45:
46: "LABEL AXES"
47: for L=V[1]+((V[2]-V[1])/10)R) to V[2]-R by R;plt H[2],L,1
48: plt H[2]-(H[2]-H[1])/100,L,2;next L
49: for L=H[2]-WD to H[1]+WD by -WD;plt L,V[2],1;plt L,V[2]-R/7,2;next L
50: csiz .7,2,1;for L=V[2] to V[1] by -R;plt H[1]-(H[2]-H[1])/12,L-R/10,1
51: if 1e9L)100;fxd 1;lbl str(L*1e9);gto +3
52: if 1e9L)10;fxd 1;lbl " "&str(L*1e9)&" ";gto +2
53: fxd 1;lbl "  "&str(L*1e9)
54: if L≠V[1] and L≠V[2];plt H[1],L,1;plt H[1]+(H[2]-H[1])/100,L,2
55: next L;csiz 1.5,2,1,90;plt H[1]-(H[2]-H[1])/12,V[1]+2.1R,1
56: lbl "Time Interval A to B in nanoseconds";csiz .7,2,1
57: for L=H[1] to H[2] by WD;if L=H[1] or L=H[2];jmp 2
58: plt L,V[1]+R/7,1;plt L,V[1],2
59: plt L-(H[2]-H[1])/50,V[1]-.35R,1;L)X;for M=1 to 12
60: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
61: if X-2764800<0;gto "PRINT"
62: X-2678400)X
63: if not (M=4 or M=6 or M=9 or M=11);jmp 3
64: if X-2678400<0;gto "PRINT"
65: X-2592000)X
66: if not M=2;jmp 3
67: if X-2505600<0;gto "PRINT"
68: X-2419200)X
69: next M
70: "PRINT";if D=86400;fxd 0;lbl str(X/D)
71: if D=3600;fxd 0;lbl str((X-86400int(X/86400))/D)
72: if D=60;fxd 0;lbl str((X-3600int(X/3600))/D)
73: if D=1;fxd 0;lbl str((X-60int(X/60))/D)
74: next L;csiz 1.5,2,1;plt H[1]+(H[2]-H[1])/4.2,V[1]-.85R,1
75: rread 1,5,R$;R$[1,2]&"-"&R$[3,4]&"  "&R$[5,6]&":"&R$[7,8]&":"&R$[9,10])M$
76: lbl "Time in "&U$&" beginning  "&M$&" "
77:
78: "PLOT DATA"
79: rread 1,5,R$,R
80: plt 'DATE',R,1;cplt -.33,-.15;lbl "o";cplt -.67,.15
81: for L=2 to Z;rread 1,L*4,R$,R
82: plt 'DATE',R,2;next L
83: pen;cplt -.33,-.15;lbl "o"
84: "RETURN";ret
85:
86: "DATE"
```

```
87: 0)X;for M=1 to val(R$[1,2])-1
88: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;X=2678400)X;jmp 3
89: if M=4 or M=6 or M=9 or M=11;X=2592000)X;jmp 2
90: if M=2;X=2419200)X
91: next M
92: X=86400val(R$[3,4])+3600val(R$[5,6]))X
93: ret X+60val(R$[7,8])+val(R$[9,10])
#28391
```

9.11 Program: PHA05

```
0: "PROGRAM:  CUMULATIVE PHASE MINUS LINEAR OFFSET PLOT";
1:
2: "*************************************************";
3: "*  Kevin Miller              *  File: PHA05       *";
4: "*  July 14, 1982             *  Update: July 14, 1982 *";
5: "*  Naval Research Laboratory *  Output Device: 7245  *";
6: "*  Space Applications Branch *  Select Code: 705    *";
7: "*************************************************";
8:
9: "GET FILE NAME AND PLOT INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PRINTER",706,"PLOTTER",705,"CLOCK",9;flt 11
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],B$[10],E$[10]
13: dim C,D,F,G,I,K,L,M,N,P,Q,V,4]K
14: ""]F$;ent "Enter Data File Name",F$;if flg13;jmp 0
15: asgn F$,1,1,0
16: if 0=1;dsp F$&" does not exist";wait 1500;jmp -2
17: asgn F$,1,1;rread 1,1,G,N,I,D$,P
18: if N<3;dsp "Not Enough Data for Plot";stp
19: ent "Plot Data from:  MMDDHHMMSS",B$
20: if flg13;"0000000000"]B$
21: if len(B$)<10;dsp "Not in Proper Form";wait 1500;jmp -2
22: ent "Plot Data to:  MMDDHHMMSS",E$
23: if flg13;"9999999999"]E$
24: if len(E$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
25:
26: "CALCULATE UNITS FOR AXES";
27: 0]W]Z;rread 1,K+1]K,T$,R;if T$<B$;jmp 0
28: "TIME"]M[1]A;R]X;for L=K+1 to N;R]S;rread 1,L,T$,R;if T$>E$;jmp 4
29: if R-S>.5P;Z-1]Z;jmp 2
30: if S-R>.5P;Z+1]Z
31: next L
32: if L#N;L-1]N
33: "TIME"]M[2]B;R-X+ZP]Y;X]R;for L=K+1 to N;R]S;rread 1,L,T$,R
34: if R-S>.5P;W-1]W;jmp 2
35: if S-R>.5P;W+1]W
36: R-X+WP-Y("TIME"-M[1])/(M[2]-M[1]))C;if abs(C)>V;abs(C)]V
37: next L;N-K]F
38: if M[2]-M[1]>86399;"DAYS"]U$;86400]D;jmp 4
39: if M[2]-M[1]>3599;"HOURS"]U$;3600]D;jmp 3
40: if M[2]-M[1]>59;"MINUTES"]U$;60]D;jmp 2
41: "SECONDS"]U$;1]D
42: Dint(M[1]/D)]M[1];Dint(M[2]/D+1)]M[2]
43: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6]]T$
44:
45: "INITIALIZE PLOT AREA";
46: fxd 0;hdcpy 1;psc 705;scl -7.125,0,0,10.5;plt 0,0,1
47: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
48: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
49: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
50: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.55,1;lbl " Cumulative Phase "
51: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
52: plt -1/16,3.52,1;lbl F$
53: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
54: plt -1/16,5.1,1;lbl str(F)
55: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
56: fxd 2;lbl str(1/P*1e-6)&" MHz "
57: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
58: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
59: plt -9/16,9.25,1;lbl "Counter Gate";plt -5.5/16,9.6,1;lbl "Time"
60: plt -1/16,9.35,1;fxd 0;lbl str(G)&" msec";plt -1.375,10.4,1
61: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4
62: plt -4.25,1,1;plt -4.25,10.4,2;flt 9;10^int(log(V)))V
63: sel 10V,-14.78261V,M[1]-(M[2]-M[1])/9.4,M[2]+(M[2]-M[1]))/94
64:
65: "LABEL AXES";
66: for L=-9V to 9V by V;if L#0;plt L,M[2],1;plt L,M[2]-(M[2]-M[1])/100,2
67: next L
68: for L=M[2]-D to M[1]+D by -D;plt 10V,L,1;plt 9.7V,L,2;next L
69: csiz 1,2,1,90;for L=10V to -10V by -V;plt L-.25V,M[1]-(M[2]-M[1])/25,1
70: if L/V<0 and L/V#-10;fxd 1;lbl " "&str(L/V);jmp 3
71: if L/V=10 or L/V=-10;fxd 1;lbl str(L/V);jmp 2
72: fxd 1;lbl " "&str(L/V)
73: if L#10V and L#-10V and L#0;plt L,M[1],1;plt L,M[1]+(M[2]-M[1])/100,2
74: next L;csiz 1.5,2,1,180;plt -6V,M[1]-(M[2]-M[1])/13,1
75: lbl "Phase Minus Linear Offset (in secs) x 10"
76: iplt .2V,(M[1]-M[2])/100,1;fxd 0;lbl str(int(log(V)));csiz 1,2,1,90
77: for L=M[1] to M[2] by D;if L=M[1] or L=M[2];jmp 2
78: plt -9.7V,L,1;plt -10V,L,2
79: plt -10.6V,L-(M[2]-M[1])/100,1;L]Q;for M=1 to 12
80: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
81: if Q-2764800<0;gto "PRINT"
82: Q-2678400]Q
83: if not (M=4 or M=6 or M=9 or M=11);jmp 3
84: if Q-2678400<0;gto "PRINT"
85: Q-2592000]Q
86: if not M-2;jmp 3
```

```
87: if Q-2505600<0;gto "PRINT"
88: Q-2419200)Q
89: next M
90: "PRINT";if D-86400;fxd 0;lbl str(Q/D)
91: if D-3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
92: if D-60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
93: if D-1;fxd 0;lbl str((Q-60int(Q/60))/D)
94: next L;csiz 1.5,2,1,90;plt -11.5V,M[1]+(M[2]-M[1])/3,1
95: if B$="0000000000";jmp 2
96: B$[1,2]&":"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M$;jmp 2
97: rread 1,5,T$;T$[1,2]&":"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
98: lbl "Time in "&U$&" beginning  "&M$
99:
100: "PLOT DATA";
101: rread 1,K,T$,R;plt 0,"TIME")T,1;cplt -.33,-.15;lbl "o";plt 0,T,1
102: 0)W;X)R;for L=K+1 to N;R)S;rread 1,L,T$,R
103: if R-S).5P;W-1)W;jmp 2
104: if S-R).5P;W+1)W
105: R-X+WP-Y(("TIME")T)-A)/(B-A))C
106: plt C,T,2;next L;cplt -.33,-.15;lbl "o";wtb "PRINTER",12,13,27,69
107: end
108:
109: "TIME";
110: 0)Q;for M=1 to val(T$[1,2])-1
111: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;jmp 3
112: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;jmp 2
113: if M=2;Q+2419200)Q
114: next M
115: Q+86400val(T$[3,4])+3600val(T$[5,6]))Q
116: ret Q+60val(T$[7,8])+val(T$[9,10])
#6142
```

9.12 Program: OFF05

```
0: "PROGRAM:  FRACTIONAL FREQUENCY OFFSET PLOT";
1:
2: "*************************************************";
3: "*  Kevin Miller                 *  File: OFF05     *";
4: "*  January 14, 1988             *  Update: December 20, 1988  *";
5: "*  Naval Research Laboratory    *  Output Device: 7245        *";
6: "*  Space Applications Branch    *  Select Code: 705           *";
7: "*************************************************";
8:
9: "GET FILE NAME AND PLOT INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PRINTER",706,"PLOTTER",705,"CLOCK",9;flt 11
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],B$[10],E$[10]
13: dim C,D,F,G,I,K,L,M,N,P,Q,V,4]K
14: ""]F$;ent "Enter Data File Name",F$;if flg13;jmp 0
15: assn F$,1,1,0
16: if Q=1;dsp F$&" does not exist";wait 1500;jmp -2
17: assn F$,1,1;rread 1,1,G,N,I,D6,P
18: if N<3;dsp "Not Enough Data for Plot";stp
19: ent "Plot Data from:  MMDDHHMMSS",B$
20: if flg13;"0000000000"]B$
21: if len(B$)<10;dsp "Not in Proper Form";wait 1500;jmp -2
22: ent "Plot Data to:  MMDDHHMMSS",E$
23: if flg13;"9999999999"]E$
24: if len(E$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
25:
26: "CALCULATE UNITS FOR AXES";
27: rread 1,K+1]K,T$,S;if T$<B$;jmp 0
28: rread 1,K+1]K,T$,R;"TIME"]H[I];1]F
29: if R-S>.5P;S+P]S;jmp 2
30: if S-R>.5P;S-P]S
31: abs((R-S)/I))V;for L=K+1 to N
32: R]S;rread 1,L,T$,R;if T$>E$;jmp 5
33: if R-S>.5P;S+P]S;jmp 2
34: if S-R>.5P;S-P]S
35: if abs((R-S)/I)>V;abs((R-S)/I))V
36: F+1]F;next L
37: rread 1,L-1,T$;"TIME"]H[2];if H[2]-H[1]>86399;"DAYS"]U$;86400]D;jmp 4
38: if H[2]-H[1]>3599;"HOURS"]U$;3600]D;jmp 3
39: if H[2]-H[1]>59;"MINUTES"]U$;60]D;jmp 2
40: "SECONDS"]U$;1]D
41: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2]
42: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])T$;4]K
43:
44: "INITIALIZE PLOT AREA";
45: fxd 0;hdcpy 1;psc 705;scl -7.125,0,0,10.5;plt 0,0,1
46: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
47: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
48: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
49: plt -7.5/16,2,1;lbl "Progrem";plt -1/16,1.55,1;lbl " Frequency Offset "
50: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
51: plt -1/16,3.52,1;lbl F$
52: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
53: plt -1/16,5.1,1;lbl str(F)
54: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
55: fxd 2;lbl str(1/P*1e-6)&" MHz "
56: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
57: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
58: plt -9/16,9.25,1;lbl "Counter Gate";plt -5.5/16,9.6,1;lbl "Time"
59: plt -1/16,9.35,1;fxd 0;lbl str(G)&" msec";plt -1.375,10.4,1
60: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4
61: plt -4.25,1,1;plt -4.25,10.4,2;flt 9;10^int(log(V)))V
62: scl 10V,-14.78261V,H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
63:
64: "LABEL AXES";
65: for L=-9V to 9V by V;if L#0;plt L,H[2],1;plt L,H[2]-(H[2]-H[1])/100,2
66: next L
67: for L=H[2]-D to H[1]+D by -D;plt 10V,L,1;plt 9.7V,L,2;next L
68: csiz 1,2,1,90;for L=10V to -10V by -V;plt L-.25V,H[1]-(H[2]-H[1])/25,1
69: if L/V<0 and L/V#-10;fxd 1;lbl " "&str(L/V);jmp 3
70: if L/V=10 or L/V=-10;fxd 1;lbl str(L/V);jmp 2
71: fxd 1;lbl " "&str(L/V)
72: if L#10V and L#-10V and L#0;plt L,H[1],1;plt L,H[1]+(H[2]-H[1])/100,2
73: next L;csiz 1.5,2,1,180;plt -5.9V,H[1]-(H[2]-H[1])/13,1
74: lbl "Fractional Frequency Offset x 10"
75: iplt .2V,(H[1]-H[2])/100,1;fxd 0;lbl str(int(log(V)));csiz 1,2,1,90
76: for L=H[1] to H[2] by D;if L=H[1] or L=H[2];jmp 2
77: plt -9.7V,L,1;plt -10V,L,2
78: plt -10.6V,L-(H[2]-H[1])/100,1;L]Q;for M=1 to 12
79: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
80: if Q-2764800<0;gto "PRINT"
81: Q-2678400]Q
82: if not (M=4 or M=6 or M=9 or M=11);jmp 3
83: if Q-2678400<0;gto "PRINT"
84: Q-2592000]Q
85: if not M=2;jmp 3
86: if Q-2505600<0;gto "PRINT"
```

```
87: Q-241920)@
88: next M
89: "PRINT":if D-86400;fxd 0;lbl str(Q/D)
90: if D-3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
91: if D-60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
92: if D-1;fxd 0;lbl str((Q-60int(Q/60))/D)
93: next L;csiz 1.5,2,1,90;plt -11.5V,M[1]+(M[2]-M[1])/3,1
94: if B$="0000000000";jmp 2
95: B$[1,2]&":"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M@;jmp 2
96: rread 1,5,T$;T$[1,2]&":"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M@
97: lbl "Time in "&U$&" beginning  "&M@
98:
99: "PLOT DATA":
100: rread 1,K+1)K,T@,S;if T@<B@;jmp @
101: rread 1,K+1)K,T@,R;if R-S).8P;S+P)S;jmp 2
102: if S-R).5P;S-P)$
103: plt -(R-S)/1,"TIME",1;cplt -.33,-.15;lbl "o";cplt -.67,.15
104: for L=K+1 to N;R)S;rread 1,L,T@,R;if T@>E@;jmp 4
105: if R-S).5P;S+P)S;jmp 2
106: if S-R).5P;S-P)$
107: plt -(R-S)/1,"TIME",2;next L
108: cplt -.33,-.15;lbl "o";wtb "PRINTER",12,13,27,69
109: end
110:
111: "TIME":
112: @)Q;for M=1 to val(T$[1,2])-1
113: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;jmp 3
114: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;jmp 2
115: if M=2;Q+241920@)@
116: next M
117: Q+86400val(T$[3,4])+3600val(T$[5,6])@
118: ret Q+60val(T$[7,8])+val(T$[9,10])
#30754
```

9.13 Program: OFF18

```
0: "PROGRAM:  FRACTIONAL FREQUENCY OFFSET PLOT";
1:
2: "*************************************************";
3: "*  Kevin Miller              *  File: OFF18       *";
4: "*  January 14, 1980          *  Update: December 20, 1980  *";
5: "*  Naval Research Laboratory *  Output Device: 1350        *";
6: "*  Space Applications Branch *  Select Code: 718           *";
7: "*************************************************";
8:
9: "GET FILE NAME AND PLOT INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "CLOCK",9;flt 11
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],B$[10],E$[18]
13: dim G,N,I,P,R,S,L,Q,E,M,D,V,K,F;4)K
14: ""F$;ent "Enter Data File Name",F$;if flg13;jmp 0
15: asgn F0,1,1,0
16: if Q=1;dsp F$&" does not exist";wait 1500;jmp -2
17: asgn F0,1,1;rread 1,1,G,N,I,D$,P
18: if N<3;dsp "Not Enough Data for Plot";stp
19: ent "Plot Data from:  MMDDHHMMSS",B$
20: if flg13;"0000000000")B$
21: if len(B$)≠10;dsp "Not in Proper Form";wait 1500;jmp -2
22: ent "Plot Data to:  MMDDHHMMSS",E$
23: if flg13;"9999999999")E$
24: if len(E$)≠10;dsp "Not in Proper Form";wait 1500;jmp -2
25:
26: "CALCULATE UNITS FOR AXES";
27: rread 1,K+1)K,T$,S;if T$<B$;jmp 0
28: rread 1,K+1)K,T$,R;"DATE")H[1];1)F
29: if R-S).5P;S+P)S;jmp 2
30: if S-R).5P;S-P)S
31: abs((R-S)/I))V;for L=K+1 to N
32: R)S;rread 1,L,T$,R;if T$>E$;jmp 5
33: if R-S).5P;S+P)S;jmp 2
34: if S-R).5P;S-P)S
35: if abs((R-S)/I)>V;abs((R-S)/I))V
36: F+1)F;next L
37: rread 1,L-1,T$;"DATE")H[2];if H[2]-H[1]>86399;"DAYS")U$;86400)D;jmp 4
38: if H[2]-H[1])3599;"HOURS")U$;3600)D;jmp 3
39: if H[2]-H[1])59;"MINUTES")U$;60)D;jmp 2
40: "SECONDS")U$;1)D
41: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2]
42: 1)N;if ((H[2]-H[1])/D)R)>30;2)N;if int(R/2)≠R/2;H[2]+D)H[2]
43: wrt "CLOCK","R";rcd "CLOCK",T$;4)K
44:
45: "INITIALIZE PLOT AREA";
46: fxd 0;hdcpy 0;psc 718;pclr;scl 0,10.5,0,7.125;plt 0,0,1
47: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
48: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
49: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today"
50: plt .21,1/11,1;lbl T$[1,2]&"-"&T$[4,5]&"-"&"80"
51: plt 1.85,7.5/16,1;lbl "Program";plt 1.6,1/11,1;lbl "Freq Offset"
52: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
53: plt 3.4,1/11,1;lbl F$
54: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl "Plotted"
55: plt 4.95,1/11,1;lbl str(F)&" "
56: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.1,1/11,1
57: fxd 2;lbl str(1/P*1e-6)&" MHz "
58: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
59: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
60: plt 9.45,9/16,1;lbl "Gate";plt 9.45,5.7/16,1;lbl "Time"
61: plt 9.1,1/11,1;fxd 0;lbl str(G)&" msec ";plt 10.4,1.375,1
62: plt 10.4,7.125,2;plt 1,7.125;plt 1,1.375;plt 10.4,1.375
63: plt 1,4.25,1;plt 10.4,4.25,2;flt 9;10^int(log(V)))V
64: scl H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94,-14.78261V,10V
65:
66: "LABEL AXES";
67: for L=-9V to 9V by V;if L≠0;plt H[2],L,1;plt H[2]-(H[2]-H[1])/100,L,2
68: next L
69: for L=H[2]-WD to H[1]+WD by -WD;plt L,10V,1;plt L,9.7V,2;next L
70: csiz .7,2,1;for L=10V to -10V by -V;plt H[1]-(H[2]-H[1])/11,L-.25V,1
71: if L/V=10 or L/V=-10;fxd 1;lbl " "&str(L/V);gto +2
72: fxd 1;lbl "  "&str(L/V)
73: if L≠-10V and L≠10V and L≠0;plt H[1],L,1;plt H[1]+(H[2]-H[1])/100,L,2
74: next L;csiz 1.5,2,1,90;plt H[1]-(H[2]-H[1])/12,-5.9V,1
75: lbl "Fractional Frequency Offset x 10"
76: iplt (H[1]-H[2])/100,.2V,1;fxd 0;lbl str(int(log(V)))
77: csiz .7,2,1;for L=H[1] to H[2] by WD;if L=H[1] or L=H[2];jmp 2
78: plt L,-9.7V,1;plt L,-10V,2
79: plt L-(H[2]-H[1])/50,-10.6V,1;L)Q;for M=1 to 12
80: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
81: if Q-2764800<0;gto "PRINT"
82: Q-2678400)Q
83: if not (M=4 or M=6 or M=9 or M=11);jmp 3
84: if Q-2678400<0;gto "PRINT"
85: Q-2592000)Q
86: if not M=2;jmp 3
```

```
87: if Q-2505600<0;gto "PRINT"
88: Q-2419200)Q
89: next M
90: "PRINT";if D-86400;fxd 0;lbl str(Q/D)
91: if D-3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
92: if D-60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
93: if D-1;fxd 0;lbl str((Q-60int(Q/60))/D)
94: next L;csiz 1.5,2,1;plt M[1]+(M[2]-M[1])/4.2,-11.5V,1
95: if B$="0000000000";jmp 2
96: B$[1,2]&"-"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M$;jmp 2
97: rread 1,5,T$;T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
98: lbl " Time in "&U$&" beginning  "&M$&" "
99:
100: "PLOT DATA";
101: rread 1,K+1)K,T$,S;if T$<B$;jmp 0
102: rread 1,K+1)K,T$,R;if R-S).8P;S+P)S;jmp 2
103: if S-R).8P;S-P)S
104: plt "DATE",(R-S)/I,1;cplt -.33,-.15;lbl "o";cplt -.67,.15
105: for L=K+1 to N;R)S;rread 1,L,T$,R;if T$>E$;jmp 4
106: if R-S).5P;S+P)S;jmp 2
107: if S-R).5P;S-P)S
108: plt "DATE",(R-S)/I,2;next L
109: cplt -.33,-.15;lbl "o"
110: end
111:
112: "DATE";
113: 0)Q;for M=1 to val(T$[1,2])-1
114: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;jmp 3
115: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;jmp 2
116: if M=2;Q+2419200)Q
117: next M
118: Q+86400val(T$[3,4])+3600val(T$[5,6]))Q
119: ret Q+60val(T$[7,8])+val(T$[9,10])
#11943
```

9.14 Program: .OFF05

```
0: "SUBPROGRAM:  FRACTIONAL FREQUENCY OFFSET PLOT"
1:
2: "*************************************************************"
3: "*  Kevin Miller                 *  File: .OFF05              *"
4: "*  August 10, 1988              *  Update: December 20, 1988 *"
5: "*  Naval Research Laboratory    *  Output Device: 7245       *"
6: "*  Space Applications Branch    *  Select Code: 705          *"
7: "*************************************************************"
8:
9:
10: "CALCULATE UNITS FOR AXES"
11: if K>3;jmp 3
12: spc 3;prt "Not Enough Data","Yet to do","FREQUENCY OFFSET"
13: prt "Plot.";spc 3;gto "RETURN"
14: rread 1,5,R$,S
15: rread 1,6,R$,R;"DATE")H[1]
16: if R-S>.8P;S+P)S;jmp 2
17: if S-R>.8P;S-P)S
18: abs((R-S)/I))V;for L=3 to K-1)Z
19: R)S;rread 1,L+4,R$,R
20: if R-S>.8P;S+P)S;jmp 2
21: if S-R>.8P;S-P)S
22: if abs((R-S)/I))V;abs((R-S)/I))V
23: next L;"DATE")H[2]
24: if H[2]-H[1]>86399;"DAYS")U$;86400)D;jmp 4
25: if H[2]-H[1]>3599;"HOURS")U$;3600)D;jmp 3
26: if H[2]-H[1]>59;"MINUTES")U$;60)D;jmp 2
27: "SECONDS")U$;1)D
28: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2]
29: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])R$
30:
31: "INITIALIZE PLOT AREA"
32: fxd 0;hdcpy 1;psc 705;pclr;scl -7.125,0,0,10.5;plt 0,0,1
33: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
34: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
35: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl R$
36: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.55,1;lbl " Frequency Offset "
37: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
38: plt -1/16,3.52,1;lbl F$
39: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
40: plt -1/16,5.1,1;fxd 0;lbl str(Z)
41: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
42: fxd 2;lbl str(1/P#1e-6)&" MHz "
43: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
44: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
45: plt -9/16,9.25,1;lbl "Counter Gate";plt -5.5/16,9.6,1;lbl "Time"
46: plt -1/16,9.35,1;fxd 0;lbl str(G)&" msec";plt -1.375,10.4,1
47: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4
48: plt -4.25,1,1;plt -4.25,10.4,2;flt 9;10^int(log(V)))V
49: scl 10V,-14.78261V,H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
50:
51: "LABEL AXES"
52: for L=-9V to 9V by V;if L<0;plt L,H[2],1;plt L,H[2]-(H[2]-H[1])/100,2
53: next L
54: for L=H[2]-D to H[1]+D by -D;plt 10V,L,1;plt 9.7V,L,2;next L
55: csiz 1,2,1,90;for L=10V to -10V by -V;plt L-.25V,H[1]-(H[2]-H[1])/25,1
56: if L/V=10 or L/V=-10;fxd 1;lbl str(L/V);jmp 2
57: fxd 1;lbl " "&str(L/V)
58: if L<10V and L>-10V and L<0;plt L,H[1],1;plt L,H[1]+(H[2]-H[1])/100,2
59: next L;csiz 1.5,2,1,180;plt -5.9V,H[1]-(H[2]-H[1])/13,1
60: lbl "Fractional Frequency Offset x 10"
61: iplt .2V,(H[1]-H[2])/100,1;fxd 0;lbl str(int(log(V)));csiz 1,2,1,90
62: for L=H[1] to H[2] by D;if L=H[1] or L=H[2];jmp 2
63: plt -9.7V,L,1;plt -10V,L,2
64: plt -10.6V,L-(H[2]-H[1])/100,1;L)X;for M=1 to 12
65: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
66: if X-2764800<0;gto "PRINT"
67: X-2678400)X
68: if not (M=4 or M=6 or M=9 or M=11);jmp 3
69: if X-2678400<0;gto "PRINT"
70: X-2592000)X
71: if not M=2;jmp 3
72: if X-2505600<0;gto "PRINT"
73: X-2419200)X
74: next M
75: "PRINT";if D=86400;fxd 0;lbl str(X/D)
76: if D=3600;fxd 0;lbl str((X-86400int(X/86400))/D)
77: if D=60;fxd 0;lbl str((X-3600int(X/3600))/D)
78: if D=1;fxd 0;lbl str((X-60int(X/60))/D)
79: next L;csiz 1.5,2,1,90;plt -11.5V,H[1]+(H[2]-H[1])/3,1
80: rread 1,6,R$;R$[1,2]&"-"&R$[3,4]&"  "&R$[5,6]&":"&R$[7,8]&":"&R$[9,10])M$
81: lbl "Time in "&U$&" beginning  "&M$
82:
83: "PLOT DATA"
84: rread 1,5,R$,S
85: rread 1,6,R$,R;if R-S>.8P;S+P)S;jmp 2
86: if S-R>.8P;S-P)S
```

```
87: plt (R-S)/1,'DATE',1;cplt -.33,-.15;lbl 'e';cplt -.67,.15
88: for L=3 to Z;R)S;rread 1,L+4,R$,R
89: if R-S).8P;S+P)S;jmp 2
90: if S-R).8P;S-P)S
91: plt (R-S)/1,'DATE',2;next L
92: cplt -.33,-.15;lbl 'e'
93: wtb 'PRINTER',10,13,27,69
94: 'RETURN';ret
95:
96: 'DATE'
97: 0)X;for M=1 to val(R$[1,2])-1
98: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;X+2678400)X;jmp 3
99: if M=4 or M=6 or M=9 or M=11;X+2592000)X;jmp 2
100: if M=2;X+2419200)X
101: next M
102: X+86400val(R$[3,4])+3600val(R$[5,6]))X
103: ret X+60val(R$[7,8])+val(R$[9,10])
#62
```

9.15 Program: .OFF18

```
0: "SUBPROGRAM:  FRACTIONAL FREQUENCY OFFSET PLOT";
1:
2: "**************************************************";
3: "*  Kevin Miller                  *  File: .OFF18    *";
4: "*  December 19, 1980             *  Update: December 20, 1980  *";
5: "*  Naval Research Laboratory     *  Output Device: 1350        *";
6: "*  Space Applications Branch     *  Select Code: 718           *";
7: "**************************************************";
8:
9: "CALCULATE UNITS FOR AXES";
10: hdcpy 0;psc 718;flt 11;if K>3;jmp 3
11: spc 3;prt "Not Enough Data","Yet to do","FREQUENCY OFFSET"
12: prt "Plot.";spc 3;gto "RETURN"
13: rread 1,5,R$,S;rread 1,6,R$,R;"DATE")M[1]
14: if R-S).8P;S+P)S;jmp 2
15: if S-R).8P;S-P)S
16: abs((R-S)/I))V;for L=3 to K-1)Z
17: R)S;rread 1,L+4,R$,R
18: if R-S).8P;S+P)S;jmp 2
19: if S-R).8P;S-P)S
20: if abs((R-S)/I))>V;abs((R-S)/I))V
21: next L;"DATE")M[2]
22: if M[2]-M[1]>86399;"DAYS")U$;86400)D;jmp 4
23: if M[2]-M[1]>3599;"HOURS")U$;3600)D;jmp 3
24: if M[2]-M[1]>59;"MINUTES")U$;60)D;jmp 2
25: "SECONDS")U$;1)D
26: Dint(M[1]/D))M[1];Dint(M[2]/D+1))M[2]
27: 1)W;if ((M[2]-M[1])/D)R)>30;2)W;if int(R/2)≠R/2;M[2]+D)M[2]
28:
29: "INITIALIZE PLOT AREA";
30: fxd 0;pclr;scl 0,10.5,0,7.125;plt 0,0,1
31: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
32: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
33: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today"
34: plt .21,1/11,1;lbl D$[1,2]&"-"&D$[4,5]&"-"&D$[5,6]&" "
35: plt 1.85,7.5/16,1;lbl "Program";plt 1.6,1/11,1;lbl "Freq Offset"
36: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
37: plt 3.4,1/11,1;lbl F$&" "
38: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl "Plotted"
39: plt 4.95,1/11,1;fxd 0;lbl str(Z-1)&" "
40: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.1,1/11,1
41: fxd 2;lbl str(1/P#1e-6)&" MHz "
42: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
43: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
44: plt 9.45,9/16,1;lbl "Gate";plt 9.45,5.7/16,1;lbl "Time"
45: plt 9.1,1/11,1;fxd 0;lbl str(G)&" msec ";plt 10.4,1.375,1
46: plt 10.4,7.125,2;plt 1,7.125;plt 1,1.375;plt 10.4,1.375
47: plt 1,4.25,1;plt 10.4,4.25,2;flt 9;10^int(log(V)))V
48: scl M[1]-(M[2]-M[1])/9.4,M[2]+(M[2]-M[1])/94,-14.78261V,10V
49:
50: "LABEL AXES";
51: for L=-9V to 9V by V;if L≠0;plt M[2],L,1;plt M[2]-(M[2]-M[1])/100,L,2
52: next L
53: for L=M[2]-WD to M[1]+WD by -WD;plt L,10V,1;plt L,9.7V,2;next L
54: csiz .7,2,1;for L=10V to -10V by -V;plt M[1]-(M[2]-M[1])/11,L-.25V,1
55: if L/V=10 or L/V=-10;fxd 1;lbl " "&str(L/V)&" ";gto +2
56: fxd 1;lbl "  "&str(L/V)
57: if L≠-10V and L≠10V and L≠0;plt M[1],L,1;plt M[1]+(M[2]-M[1])/100,L,2
58: next L;csiz 1.5,2,1,90;plt M[1]-(M[2]-M[1])/12,-5.9V,1
59: lbl "Fractional Frequency Offset x 10"
60: plt (M[1]-M[2])/100,.2V,1;fxd 0;lbl str(int(log(V)))&" "
61: csiz .7,2,1;for L=M[1] to M[2] by WD;if L=M[1] or L=M[2];jmp 2
62: plt L,-9.7V,1;plt L,-10V,2
63: plt L-(M[2]-M[1])/50,-10.6V,1;L)X;for M=1 to 12
64: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
65: if X-2764800<0;gto "PRINT"
66: X-2678400)X
67: if not (M=4 or M=6 or M=9 or M=11);jmp 3
68: if X-2678400<0;gto "PRINT"
69: X-2592000)X
70: if not M=2;jmp 3
71: if X-2505600<0;gto "PRINT"
72: X-2419200)X
73: next M
74: "PRINT";if D=86400;fxd 0;lbl str(X/D)
75: if D=3600;fxd 0;lbl str((X-86400int(X/86400))/D)
76: if D=60;fxd 0;lbl str((X-3600int(X/3600))/D)
77: if D=1;fxd 0;lbl str((X-60int(X/60))/D)
78: next L;csiz 1.5,2,1;plt M[1]+(M[2]-M[1])/4.2,-11.5V,1
79: rread 1,6,R$;R$[1,2]&"-"&R$[3,4]&"  "&R$[5,6]&":"&R$[7,8]&":"&R$[9,10]]M$
80: lbl "Time in "&U$&" beginning  "&M$&" "
81:
82: "PLOT DATA";
83: rread 1,5,R$,S;rread 1,6,R$,R
84: if R-S).8P;S+P)S;jmp 2
85: if S-R).8P;S-P)S
86: plt "DATE",(R-S)/I,1;cplt -.33,-.15;lbl "e";cplt -.67,.15
```

```
87: for L=3 to 2;R)S;rread 1,L+4,R$,R
88: if R-S).8P;S-P)S;;mp 2
89: if S-R).8P;S-P)S
90: plt 'DATE',(R-S)/1,2;next L
91: cplt -.33,-.15;lbl "e"
92: "RETURN":ret
93:
94: "DATE":
95: 0)X;for M=1 to val(R$[1,2])-1
96: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;X+2678400)X;;mp 3
97: if M=4 or M=6 or M=9 or M=11;X+2592000)X;;mp 2
98: if M=2;X+2419200)X
99: next M
100: X+86400val(R$[3,4])+3600val(R$[5,6]))X
101: ret X+60val(R$[7,8])+val(R$[9,10])
=10021
```

9.16 Program: VOFF05

```
0: "PROGRAM:  FRACTIONAL FREQUENCY OFFSET PLOT, CONSTANT OFFSET REMOVABLE";
1:
2: "*************************************************************";
3: "*  Kevin Miller                 *  File: VOFF05              *";
4: "*  January 14, 1980             *  Update: July 30, 1983     *";
5: "*  Naval Research Laboratory    *  Output Device: 7245       *";
6: "*  Space Applications Branch    *  Select Code: 705          *";
7: "*************************************************************";
8:
9: "GET FILE NAME AND PLOT INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PRINTER",706,"PLOTTER",705,"CLOCK",9;flt 11
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],B$[10],E$[10],D$[10]
13: dim C,D,F,G,I,K,L,M,N,P,Q,V;4)K;0)Q
14: ""}F$;ent "Enter Data File Name",F$;if flg13;jmp 0
15: asgn F$,1,1,0
16: if Q=1;dsp F$&" does not exist";wait 1500;jmp -2
17: asgn F$,1,1;rread 1,1,G,N,I,D$,P
18: if N<3;dsp "Not Enough Data for Plot";stp
19: ent "Plot Data from:  MMDDHHMMSS",B$
20: if flg13;"0000000000"}B$
21: if len(B$)<10;dsp "Not in Proper Form";wait 1500;jmp -2
22: ent "Plot Data to:  MMDDHHMMSS",E$
23: if flg13;"9999999999"}E$
24: if len(E$)≠10;dsp "Not in Proper Form";wait 1500;jmp -2
25: ent "Enter offset to remove",Q;-Q)Q;sfg 11
26:
27: "CALCULATE UNITS FOR AXES";
28: rread 1,K+1)K,T$,S;if T$<B$;jmp 0
29: rread 1,K+1)K,T$,R;"TIME")H[1];1)F
30: if R-S).5P;S+P)S;jmp 2
31: if S-R).5P;S-P)S
32: abs((R-S)/I-Q))V;for L=K+1 to N
33: R)S;rread 1,L,T$,R;if T$>E$;jmp 5
34: if R-S).5P;S+P)S;jmp 2
35: if S-R).5P;S-P)S
36: if abs((R-S)/I-Q)>V;abs((R-S)/I-Q))V
37: F+1)F;next L
38: rread 1,L-1,T$;"TIME")H[2];if H[2]-H[1]>86399;"DAYS")U$;86400)D;jmp 4
39: if H[2]-H[1])3599;"HOURS")U$;3600)D;jmp 3
40: if H[2]-H[1])59;"MINUTES")U$;60)D;jmp 2
41: "SECONDS")U$;1)D
42: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2]
43: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])T$;4)K
44:
45: "INITIALIZE PLOT AREA";
46: fxd 0;hdcpy 1;psc 705;scl -7.125,0,0,10.5;plt 0,0,1
47: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
48: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
49: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
50: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.55,1;lbl " Frequency Offset "
51: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
52: plt -1/16,3.52,1;lbl F$
53: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
54: plt -1/16,5.1,1;lbl str(F)
55: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
56: fxd 2;lbl str(1/Psec-6)&" MHz "
57: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
58: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
59: plt -9/16,9.25,1;lbl "Counter Gate";plt -5.5/16,9.6,1;lbl "Time"
60: plt -1/16,9.35,1;fxd 0;lbl str(G)&" msec";plt -1.375,10.4,1
61: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4
62: plt -4.25,1,1;plt -4.25,10.4,2;flt 9;10^int(log(V)))V
63: scl 10V,-14.78261V,H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
64:
65: "LABEL AXES";
66: for L=-9V to 9V by V;if L≠0;plt L,H[2],1;plt L,H[2]-(H[2]-H[1])/100,2
67: next L
68: for L=H[2]-0 to H[1]+0 by -0;plt 10V,L,1;plt 9.7V,L,2;next L
69: csiz 1,2,1,90;for L=10V to -10V by -V;plt L-.25V,H[1]-(H[2]-H[1])/25,1
70: if L/V<0 and L/V≠-10;fxd 1;lbl " "&str(L/V);jmp 3
71: if L/V=10 or L/V=-10;fxd 1;lbl str(L/V);jmp 2
72: fxd 1;lbl " "&str(L/V)
73: if L≠10V and L≠-10V and L≠0;plt L,H[1],1;plt L,H[1]+(H[2]-H[1])/100,2
74: next L;csiz 1.5,2,1,180;plt -5.9V,H[1]-(H[2]-H[1])/13,1
75: lbl "Fractional Frequency Offset x 10"
76: plt .2V,(H[1]-H[2])/100,1;fxd 0;lbl str(int(log(V)));csiz 1,2,1,90
77: csiz 1.5,2,1,90;if flg11≠1;jmp 2
78: plt 9V,H[1]+(H[2]-H[1])/3,1;flt 2;lbl str(-Q)&" offset removed"
79: for L=H[1] to H[2] by D;if L=H[1] or L=H[2];jmp 2
80: plt -9.7V,L,1;plt -10V,L,2
81: plt -10.6V,L-(H[2]-H[1])/100,1;L)Q;for M=1 to 12
82: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
83: if Q-2764800<0;gto "PRINT"
84: Q-2678400)Q
85: if not (M=4 or M=6 or M=9 or M=11);jmp 3
86: if Q-2678400<0;gto "PRINT"
```

```
87: 0-2592000)@
88: if not M-2;jmp 3
89: if @-2505600<0;gto "PRINT"
90: @-241920@)@
91: next M
92: "PRINT";if D-86400;fxd 0;lbl str(Q/D)
93: if D-3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
94: if D-60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
95: if D-1;fxd 0;lbl str((Q-60int(Q/60))/D)
96: next L;calx 1.5,2,1,90;plt -11.5V,H(1)+(H(2)-H(1))/3,1
97: if B$-"0000000000";jmp 2
98: B$[1,2]&"-"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M@;jmp 2
99: rread 1,5,T$;T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M@
100: lbl "Time in "&U$&" beginning  "&M@
101:
102: "PLOT DATA";
103: rread 1,K+1)K,T$,5;if T$<B$;jmp @
104: rread 1,K+1)K,T$,R;if R-S).8P;S+P)S;jmp 2
105: if S-R).5P;S-P)$
106: plt -(R-S)/I-@,"TIME",1;cplt -.33,-.15;lbl "o";cplt -.67,.15
107: for L-K+1 to N;R)S;rread 1,L,T$,R;if T$>E$;jmp 4
108: if R-S).5P;S+P)S;jmp 2
109: if S-R).5P;S-P)$
110: plt -(R-S)/I-@,"TIME",2;next L
111: cplt -.33,-.15;lbl "o";wtb "PRINTER",12,13,27,69
112: end
113:
114: "TIME";
115: @)Q;for M-1 to val(T$[1,2])-1
116: if M-1 or M-3 or M-5 or M-7 or M-8 or M-10 or M-12;Q+2678400)Q;jmp 3
117: if M-4 or M-6 or M-9 or M-11;Q+2592000)Q;jmp 2
118: if M-2;Q+241920@)@
119: next M
120: Q+86400val(T$[3,4])+3600val(T$[5,6]))@
121: ret Q+60val(T$[7,8])+val(T$[9,10])
*26705
```

9.17 Program: VOFF18

```
0: "PROGRAM:  FRACTIONAL FREQUENCY OFFSET PLOT, CONSTANT OFFSET REMOVABLE";
1:
2: "ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß";
3: "ß  Kevin Miller                ß  File: VOFF18                    ß";
4: "ß  January 14, 1988            ß  Update: July 30, 1983           ß";
5: "ß  Naval Research Laboratory   ß  Output Device: 1350             ß";
6: "ß  Space Applications Branch   ß  Select Code: 718                ß";
7: "ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß";
8:
9: "GET FILE NAME AND PLOT INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "CLOCK",9;flt 11
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],B$[10],E$[10],O$[10]
13: dim G,M,I,P,R,S,L,Q,E,M,D,V,K,F;4)K;8)O
14: ""}F$;ent "Enter Data File Name",F$;if flg13;jmp 8
15: assn F$,1,1,Q
16: if Q=1;dsp F$&" does not exist";wait 1500;jmp -2
17: assn F$,1,1;rread 1,1,G,M,I,D$,P
18: if M<3;dsp "Not Enough Data for Plot";stp
19: ent "Plot Data from:  MMDDHHMMSS",B$
20: if flg13;"0000000000"}B$
21: if len(B$)≠10;dsp "Not in Proper Form";wait 1500;jmp -2
22: ent "Plot Data to:  MMDDHHMMSS",E$
23: if flg13;"9999999999"}E$
24: if len(E$)≠10;dsp "Not in Proper Form";wait 1500;jmp -2
25: ent "Enter offset to remove",O;-O}O;sfg 11
26:
27: "CALCULATE UNITS FOR AXES";
28: rread 1,K+1}K,T$,S;if T$<B$;jmp 0
29: rread 1,K+1}K,T$,R;"DATE"}H[1];1}F
30: if R-S).5P;S+P}S;jmp 2
31: if S-R).5P;S-P}S
32: abs((R-S)/I-O)}V;for L=K+1 to M
33: R}S;rread 1,L,T$,R;if T$>E$;jmp 5
34: if R-S).5P;S+P}S;jmp 2
35: if S-R).5P;S-P}S
36: if abs((R-S)/I-O)>V;abs((R-S)/I-O)}V
37: F+1}F;next L
38: rread 1,L-1,T$;"DATE"}H[2];if H[2]-H[1]>86399;"DAYS"}U$;86400}D;jmp 4
39: if H[2]-H[1]>3599;"HOURS"}U$;3600}D;jmp 3
40: if H[2]-H[1]>59;"MINUTES"}U$;60}D;jmp 2
41: "SECONDS"}U$;1}D
42: Dint(H[1]/D)}H[1];Dint(H[2]/D+1)}H[2]
43: 1}W;if ((H[2]-H[1])/D)R)>30;2}W;if int(R/2)≠R/2;H[2]+D}H[2]
44: wrt "CLOCK","R";red "CLOCK",T$;4)K
45:
46: "INITIALIZE PLOT AREA";
47: fxd 0;hdcpy 0;psc 718;plr;scl 0,10.5,0,7.125;plt 0,0,1
48: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
49: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
50: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today"
51: plt .21,1/11,1;lbl T$[1,2]&"-"&T$[4,5]&"-"&"80"
52: plt 1.85,7.5/16,1;lbl "Program";plt 1.6,1/11,1;lbl "Freq Offset"
53: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
54: plt 3.4,1/11,1;lbl F$
55: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl "Plotted"
56: plt 4.95,1/11,1;lbl str(F)&" "
57: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.1,1/11,1
58: fxd 2;lbl str(1/P#1e-6)&" MHz "
59: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
60: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
61: plt 9.45,9/16,1;lbl "Gate";plt 9.45,5.7/16,1;lbl "Time"
62: plt 9.1,1/11,1;fxd 0;lbl str(G)&" msec ";plt 10.4,1.375,1
63: plt 10.4,7.125,2;plt 1,7.125;plt 1,1.375;plt 10.4,1.375
64: plt 1,4.25,1;plt 10.4,4.25,2;flt 9;10^int(log(V))V
65: scl H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94,-14.78261V,10V
66:
67: "LABEL AXES";
68: for L=-9V to 9V by V;if L≠0;plt H[2],L,1;plt H[2]-(H[2]-H[1])/100,L,2
69: next L
70: for L=H[2]-WD to H[1]+WD by -WD;plt L,10V,1;plt L,9.7V,2;next L
71: csiz .7,2,1;for L=10V to -10V by -V;plt H[1]-(H[2]-H[1])/11,L-.25V,1
72: if L/V=10 or L/V=-10;fxd 1;lbl " "&str(L/V);gto "2
73: fxd 1;lbl "  "&str(L/V)
74: if L≠-10V and L≠10V and L≠0;plt H[1],L,1;plt H[1]+(H[2]-H[1])/100,L,2
75: next L;csiz 1.5,2,1,90;plt H[1]-(H[2]-H[1])/12,-5.9V,1
76: lbl "Fractional Frequency Offset x 10"
77: iplt (H[1]-H[2])/100,.2V,1;fxd 0;lbl str(int(log(V)))
78: csiz 1.5,2,1,0;if flg11≠1;jmp 2
79: plt H[1]+(H[2]-H[1])/4.2,9V,1;flt 2;lbl str(-O)&" offset removed"
80: csiz .7,2,1;for L=H[1] to H[2] by WD;if L=H[1] or L=H[2];jmp 2
81: plt L,-9.7V,1;plt L,-10V,2
82: plt L-(H[2]-H[1])/50,-10.6V,1;L}Q;for M=1 to 12
83: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
84: if Q-2764800<0;gto "PRINT"
85: Q-2678400}Q
86: if not (M=4 or M=6 or M=9 or M=11);jmp 3
```

```
87: if Q-2678400<0;gto "PRINT"
88: Q-2592000)Q
89: if not M-2;jmp 3
90: if Q-2505600<0;gto "PRINT"
91: Q-2419200)Q
92: next M
93: "PRINT":if D-86400;fxd 0;lbl str(Q/D)
94: if D-3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
95: if D-60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
96: if D-1;fxd 0;lbl str((Q-60int(Q/60))/D)
97: next L;csiz 1.5,2,1;plt M[1]+(M[2]-M[1])/4.2,-11.5V,1
98: if B$-"0000000000";jmp 2
99: B$[1,2]&"-"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10]]M$;jmp 2
100: rread 1,5,T$;T$[1,2]&"-"&T$.3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10]]M$
101: lbl " Time in "&U$&" beginning  "&M$&" "
102:
103: "PLOT DATA":
104: rread 1,K+1)K,T$,S;if T$<B$;jmp 0
105: rread 1,K+1)K,T$,R;if R-S).8P;S+P)S;jmp 2
106: if S-R).8P;S-P)S
107: plt 'DATE',-(R-S)/1-0,1;cplt -.33,-.15;lbl "e";cplt -.67,.15
108: for L-K+1 to N;R)S;rread 1,L,T$,R;if T$)E$;jmp 4
109: if R-S).5P;S+P)S;jmp 2
110: if S-R).5P;S-P)S
111: plt 'DATE',-(R-S)/1-0,2;next L
112: cplt -.33,-.15;lbl "e"
113: end
114:
115: "DATE":
116: 0)Q;for M-1 to val(T$[1,2])-1
117: if M-1 or M-3 or M-5 or M-7 or M-8 or M-10 or M-12;Q+2678400)Q;jmp 3
118: if M-4 or M-6 or M-9 or M-11;Q+2592000)Q;jmp 2
119: if M-2;Q+2419200)Q
120: next M
121: Q+86400val(T$[3,4])+3600val(T$[5,6])Q
122: ret Q+60val(T$[7,8])+val(T$[9,10])
#6589
```

9.18 Program: AVG05

```
"PROGRAM:  OFFSET FREQUENCY AVERAGE PLOT"
!
"*************************************************************"
"*  Kevin Miller                 *  File: AVG05               *"
"*  July 18, 1982                *  Update: July 18, 1982     *"
"*  Naval Research Laboratory    *  Output Device: 7245       *"
"*  Space Applications Branch    *  Select Code: 705          *"
"*************************************************************"
!
"GET FILE NAME AND PLOT INTERVAL"
dsp "Insert Softcopy Graphics Tape";stp
ldb 1;dev "PRINTER",706,"PLOTTER",705,"CLOCK",9;flt 11
dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],B$[10],E$[10]
dim A,B,C,D,E,F,G,I,K,L,M,N,P,Q,R,S,U,V,W,X,Y,Z;4)K
""]F$;ent "Enter Data File Name",F$;if flg13;jmp 0
assn F$,1,1,0
if Q=1;dsp F$&" does not exist";wait 1500;jmp -2
assn F$,1,1;rread 1,1,G,N,I,D$,P
if N<3;dsp "Not Enough Data for Plot";stp
!
"CALCULATE UNITS FOR AXES"
rread 1,K<1)K,T$,S
rread 1,K+1)K,T$,R;"TIME")H[1];1)F
if R-S).5P;S+P)S;jmp 2
if S-R).5P;S-P)S
abs((R-S)/I))V;for L=K+1 to N
R)S;rread 1,L,T$,R
if R-S).5P;S+P)S;jmp 2
if S-R).5P;S-P)S
if abs((R-S)/I))V;abs((R-S)/I))W
F+1)F;next L
rread 1,L-1,T$;"TIME")H[2];if H[2]-H[1]>86399;"DAYS")U$;86400)D;jmp 4
if H[2]-H[1]>3599;"HOURS")U$;3600)D;jmp 3
if H[2]-H[1]>59;"MINUTES")U$;60)D;jmp 2
"SECONDS")U$;1)D
Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2]
D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])T$;4)K
!
"INITIALIZE PLOT AREA"
fxd 0;hdcpy 1;psc 705;scl -7.125,0,0,10.5;plt 0,0,1
plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
plt -7.5/16,2,1;lbl "Program";plt -1/16,1.55,1;lbl " Offset Freq Avg  "
plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
plt -1/16,3.52,1;lbl F$
plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
fxd 2;lbl str(1/P#1e-6)&" MHz "
plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
plt -9/16,9.25,1;lbl "Counter Gate";plt -5.5/16,9.6,1;lbl "Time"
plt -1/16,9.35,1;fxd 0;lbl str(Q)&" msec";plt -1.375,10.4,1
plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4
plt -4.25,1,1;plt -4.25,10.4,2;flt 9;10^int(log(V)))V
scl 10V,-14.78261V,H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
!
"LABEL AXES"
for L=-9V to 9V by V;if L#0;plt L,H[2],1;plt L,H[2]-(H[2]-H[1])/100,2
next L
for L=H[2]-D to H[1]+D by -D;plt 10V,L,1;plt 9.7V,L,2;next L
csiz 1,2,1,90;for L=10V to -10V by -V;plt L-.25V,H[1]-(H[2]-H[1])/25,1
if L/V<0 and L/V#-10;fxd 1;lbl " "&str(L/V);jmp 3
if L/V=10 or L/V=-10;fxd 1;lbl str(L/V);jmp 2
fxd 1;lbl " "&str(L/V)
if L#10V and L#-10V and L#0;plt L,H[1],1;plt L,H[1]+(H[2]-H[1])/100,2
next L;csiz 1.5,2,1,180;plt -5.9V,H[1]-(H[2]-H[1])/13,1
lbl "Fractional Frequency Offset x 10"
iplt .2V,(H[1]-H[2])/100,1;fxd 0;lbl str(int(log(V)));csiz 1,2,1,90
for L=H[1] to H[2] by D;if L=H[1] or L=H[2];jmp 2
plt -9.7V,L,1;plt -10V,L,2
plt -10.6V,L-(H[2]-H[1])/100,1;L)Q;for M=1 to 12
if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
if Q-2764800<0;gto "PRINT"
Q-2678400)Q
if not (M=4 or M=6 or M=9 or M=11);jmp 3
if Q-2678400<0;gto "PRINT"
Q-2592000)Q
if not M=2;jmp 3
if Q-2505600<0;gto "PRINT"
Q-2419200)Q
next M
"PRINT";if D=86400;fxd 0;lbl str(Q/D)
if D=3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
if D=60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
if D=1;fxd 0;lbl str((Q-60int(Q/60))/D)
next L;csiz 1.5,2,1,90;plt -11.5V,H[1]+(H[2]-H[1])/3,1
```

```
87: rread 1,5,T$,T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
88: lbl "Time is "&U$&" beginning  "&M$
89:
90: "PLOT DATA":
91: spc 3;prt "----------------","OFFSET FREQUENCY","   AVERAGING    "
92: prt "     PROGRAM      ","----------------";spc 2;4)K;1)U;0)A
93: ent "Start Averaging:    MMDDHHMMSS",B$
94: if not flg13;jmp 3
95: scl -7.125,0,8,10.5;plt -1/16,5.1,1;lbl str(A)
96: wtb "PRINTER",12,13,27,69;spc 2;end
97: if len(B$)#10;dsp "Not in Proper Form";wait 1500;jmp -4
98: rread 1,K+1)K,T$,S;if T$<B$;jmp 0
99: K)B
100: ent "Stop Averaging:     MMDDHHMMSS",E$
101: if len(E$)#10;dsp "Not in Proper Form";wait 1500;jmp -1
102: if E$<B$;dsp "Invalid Entry";wait 1500;jmp -2
103: rread 1,K+1)K,T$,R;if T$<E$;jmp 0
104: K)E
105: if B)E-2;dsp "Invalid Entry";wait 1500;jmp -10
106: 0)Z;rread 1,B,T$,S;S)R)X;"TIME")Y
107: for M=B+1 to E;R)S;rread 1,M,T$,R
108: if R-S).5P;Z-1)Z;jmp 2
109: if S-R).5P;Z+1)Z
110: next M;"TIME")W;rread 1,int((E+B)/2),T$
111: plt (R-X+ZP)/(W-Y),"TIME",U;pen;cplt -.33,-.15;lbl "e"
112: plt (R-X+ZP)/(W-Y),"TIME",1,2)U
113: fmt 1,c,3z,e12.5;wrt 16,1,"e",(R-X+ZP)/(W-Y);spc 2
114: A+1)A;gto 93
115: end
116:
117: "TIME":
118: 0)Q;for M=1 to val(T$[1,2])-1
119: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;jmp 3
120: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;jmp 2
121: if M=2;Q+2419200)Q
122: next M
123: Q+86400val(T$[3,4])+3600val(T$[5,6])Q
124: ret Q+60val(T$[7,8])+val(T$[9,10])
#2169
```

9.19 Program: FVT05

```
0: "PROGRAM:  FRACTIONAL FREQUENCY OFFSET vs TEMPERATURE";
1:
2: "*************************************************";
3: "*  Kevin Miller              *  File: FVT05       *";
4: "*  August 23, 1980           *  Update: December 20, 1980  *";
5: "*  Naval Research Laboratory *  Output Device: 7245       *";
6: "*  Space Applications Branch *  Select Code: 705          *";
7: "*************************************************";
8:
9: "GET FILE NAME AND PLOT INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PRINTER",706,"PLOTTER",705,"CLOCK",9;flt 11;hdcpy 1;psc 705
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],B$[10],E$[10],C$[25,10],J$[25]
13: dim H[2],C[25],C,D,F,G,I,K,L,M,N,P,Q,V,4)K
14: ent "Enter Data File Name",F$
15: asgn F$,1,1,@
16: if Q=1;dsp F$&" does not exist";wait 1500;jmp -2
17: asgn F$,1,1;rread 1,1,G,N,I,D$,P,J
18: for L=1 to J;rread 1,C[L],C$[L],J$[L,L];next L
19: if J=0;dsp "No Temperature Data Collected";stp
20: if N<3;dsp "Not Enough Data for Plot";stp
21: spc 3;prt "----------------","VOLTAGE CHANNELS","----------------";spc 2
22: prt "No:   Name:","####  ########";fxd 0
23: for L=1 to J;str(C[L])}U$;if C[L]<10;" 0"&U$[2,2]}U$
24: prt U$&"   "&C$[L];next L;spc 3
25: ent "Enter Temperature Channel Number",W
26: for L=1 to J;if C[L]=W;L}W;dim A[N,2];gto +2
27: next L;dsp "Invalid Entry";wait 1500;gto -2
28: prt "Typical Values:"," ";fxd 3
29: for L=5 to 6;rread 1,L,T$,R;for M=1 to W;rread 1,T;next M
30: prt "   "&str(T);next L;spc 3
31: ent "Enter Temperature Scale Factor",B
32: ent "Plot Data from:  MMDDHHMMSS",B$
33: if flg13;"0000000000"}B$
34: if len(B$)<10;dsp "Not in Proper Form";wait 1500;jmp -2
35: ent "Plot Data to:  MMDDHHMMSS",E$
36: if flg13;"9999999999"}E$
37: if len(E$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
38:
39: "CALCULATE UNITS FOR AXES";
40: dsp "Reading Data From Disk"
41: rread 1,K+1}K,T$,S;if T$<B$;jmp 0
42: 1}F;rread 1,K+1}K,T$,R;for M=1 to W;rread 1,T;next M;BT}A[F,1]
43: 1}F;if R-S>.8P;S+P}S;jmp 2
44: if S-R>.8P;S-P}S
45: abs((R-S)/I)A[1,2]}}V
46: for L=K+1 to N;R}S;rread 1,L,T$,R;if T$}E$;gto +6
47: F+1}F;for M=1 to W;rread 1,T;next M;BT}A[F,1]
48: if R-S>.8P;S+P}S;jmp 2
49: if S-R>.8P;S-P}S
50: if abs((R-S)/I)A[F,2]}}V;abs((R-S)/I)}}V
51: next L
52:
53: "SORT DATA INTO ASCENDING ORDER";
54: fxd 0;dsp "Sorting"&str(F)&" Data Points"
55: for L=1 to F-1;A[L,1]}M
56: for M=L+1 to F;if A[M,1]<A[L,1];A[M,1]}A[L,1];M}A[M,1]
57: next M;next L
58:
59: "MORE CONVERSATION";
60: fxd 2;prt "Temperature","Limits"," "," Low:     "&str(A[1,1])
61: prt " High:     "&str(A[F,1]);spc 3
62: -999}L;ent "Enter New Lower Limit",L
63: if L>A[F,1];dsp "Invalid Entry";wait 1500;jmp -1
64: 999}M;ent "Enter New Upper Limit",M
65: if M<A[1,1];dsp "Invalid Entry";wait 1500;jmp -1
66: for R=1 to F-1;if A[R,1]>L;jmp 2
67: next R
68: F-R+1}F;for S=1 to F-1;A[S+R-1,1]}A[S,1];next S
69: for R=2 to F;if A[R,1]>M;jmp 2
70: next R;jmp 2
71: R-1}F
72: int(A[1,1])}H[1];int(A[F,1]+1)}H[2]
73:
74: "INITIALIZE PLOT AREA";
75: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6]}T$
76: fxd 0;pelr;scl -7.125,0,0,10.5;plt 0,0,1
77: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
78: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
79: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
80: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.55,1;lbl "Freq Offset vs Tmp"
81: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
82: plt -1/16,3.52,1;lbl F$
83: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
84: plt -1/16,5.1,1;lbl str(F)
85: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
86: fxd 2;lbl str(1/P#1e-6)&" MHz "
```

```
87: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
88: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
89: plt -9/16,9.25,1;lbl "Counter Gate";plt -5.5/16,9.6,1;lbl "Time"
90: plt -1/16,9.35,1;fxd 0;lbl str(G)&" msec";plt -1.375,10.4,1
91: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4
92: plt -4.25,1,1;plt -4.25,10.4,2;flt 9;10^int(log(V)))V
93: scl 10V,-14.782G1V,H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
94:
95: "LABEL AXES",
96: for L=-9V to 9V by V;if L≠0;plt L,H[2],1;plt L,H[2]-(H[2]-H[1])/100,2
97: next L
98: for L=H[2]-1 to H[1]+1 by -1;plt 10V,L,1;plt 9.7V,L,2;next L
99: csiz 1.2,1,90;for L=10V to -10V by -V;plt L-.25V,H[1]-(H[2]-H[1])/25,1
100: if L/V<0 and L/V≠-10;fxd 1;lbl " "&str(L/V);jmp 3
101: if L/V=10 or L/V=-10;fxd 1;lbl str(L/V);jmp 2
102: fxd 1;lbl " "&str(L/V)
103: if L≠10V and L≠-10V and L≠0;plt L,H[1],1;plt L,H[1]+(H[2]-H[1])/100,2
104: next L;csiz 1.5,2,1,180;plt -5.9V,H[1]-(H[2]-H[1])/13,1
105: lbl "Fractional Frequency Offset x 10"
106: ipit .2V,(H[1]-H[2])/100,1;fxd 0;lbl str(int(log(V)));csiz 1,2,1,90
107: for L=H[1] to H[2];if L=H[1] or L=H[2];jmp 2
108: plt -9.7V,L,1;plt -10V,L,2
109: plt -10.6V,L-(H[2]-H[1])/100,1;fxd 0;lbl str(L);next L
110: csiz 1.5,2,1,90;plt -11.5V,H[1]+(H[2]-H[1])/3.3,1
111: if B$="0000000000";jmp 2
112: B$[1,2]&"-"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M$;jmp 2
113: rread 1,5,T$;T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
114: lbl "Temperature in deg C beginning  "&M$
115:
116:
117: "PLOT DATA",
118: plt A[1,2],A[1,1],1;cplt -.33,-.15;lbl "e";cplt -.67,.15
119: for L=2 to F;plt A[L,2],A[L,1],2;next L;cplt -.33,-.15;lbl "e"
120: end
#230
```

9.20 Program: SIG05

```
0: "PROGRAM:  ALLAN VARIANCE PLOT":
1:
2: "***************************************************************":
3: "*  Kevin Miller                  *  File: SIG05                *":
4: "*  January 14, 1980              *  Update: December 20, 1980  *":
5: "*  Naval Research Laboratory     *  Output Device: 7245        *":
6: "*  Space Applications Branch     *  Select Code: 705           *":
7: "***************************************************************":
8:
9: "GET FILE NAME AND PLOT INTERVAL":
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PLOTTER",705,"PRINTER",706,"CLOCK",9;flt 5
12: dim M[8],R$[3],F$[6],D$[6],M$[4],T$[16],F[2],H[2],V[2],B$[10],E$[10]
13: dim A,B,C,D,F,G,J,K,L,M,N,P,Q,R,Z
14: ""→F$;ent "Enter Data File Name",F$;if flg13;jmp 0
15: asgn F$,1,1,Q;if Q=1;dsp F$&" does not exist";wait 1500;jmp -1
16: asgn F$,1,1;rread 1,1,G,N,I,D$,P;int(log(I))→D
17: dim S[N-1];4→J;if N>=6;jmp 3
18: spc 3;fxd 0;prt "Only"&str(N)&" data","points in this","file. Must have"
19: prt "at least 6 to","do SIGMA plot.";spc 3;stp
20: ent "Plot Data from:  MMDDHHMMSS",B$
21: if flg13;"0000000000"→B$
22: if len(B$)≠10;dsp "Not in Proper Form";wait 1500;jmp -2
23: ent "Plot Data to:  MMDDHHMMSS",E$
24: if flg13;"9999999999"→E$
25: if len(E$)≠10;dsp "Not in Proper Form";wait 1500;jmp -2
26:
27: "READ AND STORE INTERVAL DATA":
28: rread 1,J+1→J,T$,A;if T$<B$;jmp 0
29: 0→C→F;for L=J+1 to N;rread 1,L,T$,B;if T$>E$;jmp 5
30: if B-A>.5P;(B-P-A)/I→S[L-J];jmp 3
31: if A-B>.5P;(B+P-A)/I→S[L-J];jmp 2
32: (B-A)/I→S[L-J]
33: C+S[L-J]^2→C;B→A;F+1→F;next L
34:
35: "INITIALIZE PLOT AREA":
36: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6]→T$
37: fxd 0;hdcpy 1;psc 705;scl -7.125,0,0,10.5;plt 0,0,1
38: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
39: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
40: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
41: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.55,1;lbl "  Allan Variance"
42: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
43: plt -1/16,3.52,1;lbl F$
44: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl " Found"
45: plt -1/16,5.1,1;lbl str(F)
46: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
47: fxd 2;lbl str(1/P*1e-6)&" MHz"
48: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
49: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
50: plt -9/16,9.25,1;lbl "Counter Gate";plt -5.5/16,9.6,1;lbl "Time"
51: plt -1/16,9.35,1;fxd 0;lbl str(G)&" msec";plt -1.375,10.4,1
52: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4;flt 9
53: 1→H[1];6→H[2];-15→V[1];-9→V[2];4→J
54: scl V[2],1.23913V[1]-.23913V[2],H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
55:
56: "LABEL AXES":
57: for L=V[1]+1 to V[2]-1;plt L,H[2],1;plt L,H[2]-.083,2;next L
58: for L=H[2]-1 to H[1]+1 by -1;plt V[2],L,1;plt V[2]-.12,L,2;next L
59: csiz 1,2,1,90;for L=V[2] to V[1] by -1;plt L-.144,H[1]-.25,1;lbl "10"
60: plt L-.06,H[1]-.167,1;fxd 0;lbl str(L)
61: if L≠V[1] and L≠V[2];plt L,H[1],1;plt L,H[1]+.083,2
62: next L;csiz 1.5,2,1,180;plt V[1]+1.8,H[1]-.417,1
63: lbl "Allan Variance,  Sigma (tau)";csiz 1,2,1,90
64: for L=H[1] to H[2] by 1;if L=H[1] or L=H[2];jmp 2
65: plt V[1]+.12,L,1;plt V[1],L,2
66: plt V[1]-.3,L-.117,1;lbl "10";plt V[1]-.168,L-.05,1;lbl str(L);next L
67: csiz 1.5,2,1,90;plt V[1]-.52,H[1]+1.9,1
68: lbl "Sample Time (tau) in seconds"
69:
70: "CALCULATE SIGMA VALUES, ERROR BARS, AND PLOT":
71: spc 3;prt " Allan Variance"," "," Sigma     Tau","********** *****"
72: 1→M[1];2→M[2];4→M[3];8→M[4];16→M[5];32→M[6];64→M[7];128→M[8]
73: for R=8 to 1 by -1;if int(F/4M[R])<=0;next R
74: 0→A;for K=1 to 4M[R];A+(S[K+1]-S[K])^2→A;next K
75: √(.5A/(4M[R]-1))→C
76: √(12M[R]-4)/(4M[R]-1)→E
77: plt log(C+EC),log(I)-.008,1;plt log(C+EC),log(I)+.008,2
78: plt log(C+EC),log(I),1;plt log(C-EC),log(I),2;plt log(C-EC),log(I)-.008,1
79: plt log(C-EC),log(I)+.008,2;plt log(C),log(I),1
80: fmt 9,e9.3,x,f6.0;wrt 16.9,C,I
81: for L=2 to R;0→A→F[1]
82: for Z=1 to M[L];F[1]+S[Z]→F[1];next Z
83: F[1]/M[L]→F[1]
84: for K=M[L] to 4M[R]-M[L] by M[L];0→F[2]
85: for Z=1 to M[L];F[2]+S[K+Z]→F[2];next Z
86: F[2]/M[L]→F[2];A+(F[2]-F[1])^2→A;F[2]→F[1];next K
```

```
87: \(.5A/(4M[R]/M[L]-1))}C
88: \(12M[R]/M[L]-4)/(4M[R]/M[L]-1)}E
89: plt log(C),log(IM[L]),2;plt log(C+EC),log(IM[L])-.008,1
90: plt log(C+EC),log(IM[L])+.008,2;plt log(C+EC),log(IM[L]),1
91: plt log(C-EC),log(IM[L]),2;plt log(C-EC),log(IM[L])-.008,1
92: plt log(C-EC),log(IM[L])+.008,2;plt log(C),log(IM[L]),1
93: wrt 16.9,C,IM[L]
94: next L;spc 4;wtb "PRINTER",12,13,27,69
95: end
#22736
```

9.21 Program: SIG13

```
0: "PROGRAM:  ALLAN VARIANCE PLOT"
1:
2: "*************************************************************"
3: "*  Kevin Miller                *  File: SIG10                 *"
4: "*  January 14, 1980            *  Update: December 20, 1980   *"
5: "*  Naval Research Laboratory   *  Output Device: 1350         *"
6: "*  Space Applications Branch   *  Select Code: 718            *"
7: "*************************************************************"
8:
9: "GET FILE NAME AND PLOT INTERVAL"
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "CLOCK",9;flt 5
12: dim M[8],R$[3],F$[6],M$[6],T$[16],F[2],H[2],V[2],B$[10],E$[10],D$[6]
13: dim A,B,C,D,F,G,J,K,L,M,N,P,Q,R,Z
14: "":0)F$;ent "Enter Data File Name",F$;if flg13;jmp 0
15: asgn F$,1,1,0;if Q=1;dsp F$&" does not exist";wait 1500;jmp -1
16: asgn F$,1,1;rread 1,1,G,N,I,D$,P;int(log(I)))D;dim SIM-1],4)J
17: if N<6;spc 3;prt "Not Enough Data","to do SIGMA","Plot.";spc 3;end
18: ent "Plot Data from:  MMDDHHMMSS",B$
19: if flg13;"0000000000"}B$
20: if len(B$)≠10;dsp "Not in Proper Form";wait 1500;jmp -2
21: ent "Plot Data to:  MMDDHHMMSS",E$
22: if flg13;"9999999999"}E$
23: if len(E$)≠10;dsp "Not in Proper Form";wait 1500;jmp -2
24:
25: "READ AND STORE INTERVAL DATA"
26: rread 1,J+1}J,T$,A;if T$<B$;jmp 0
27: 0)C)F;for L=J+1 to N;rread 1,L,T$,B;if T$>E$;jmp 5
28: if B-A).5P;(B-P-A)/I}S[L-J];jmp 3
29: if A-B).5P;(B+P-A)/I}S[L-J];jmp 2
30: (B-A)/I}S[L-J]
31: C+S[L-J]^2)C;B}A;F+1}F;next L
32:
33: "INITIALIZE PLOT AREA"
34: fxd 0;hdcpy 0;psc 718;pclr;scl 0,10.5,0,7.125;plt 0,0,1
35: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
36: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
37: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today"
38: plt .21,1/11,1;lbl D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6]
39: plt 1.85,7.5/16,1;lbl "Program";plt 1.6,1/11,1;lbl " Allan Var"
40: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
41: plt 3.4,1/11,1;lbl F$&" "
42: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl " Found"
43: plt 4.95,1/11,1;lbl str(F)&" "
44: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.15,1/11,1
45: fxd 2;lbl str(1/P*1e-6)&" MHz "
46: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
47: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
48: plt 9.45,9/16,1;lbl "Gate";plt 9.45,5.7/16,1;lbl "Time"
49: plt 9.1,1/11,1;fxd 0;lbl str(G)&" msec ";plt 10.4,1.375,1
50: plt 10.4,7.125,2;plt 1,7.125;plt 1,1.375;plt 10.4,1.375;flt 9
51: 1}M[1];6}M[2];-15}V[1];-9}V[2]
52: scl M[1]-(M[2]-M[1])/9.4,M[2]+(M[2]-M[1])/94,1.23913V[1]-.23913V[2],V[2]
53:
54: "LABEL AXES"
55: for L=V[1]+1 to V[2]-1;plt M[2],L,1;plt M[2]-.08,L,2;next L
56: for L=M[2]-1 to M[1]+1 by -1;plt L,V[2],1;plt L,V[2]-.12,2;next L
57: csiz .7,2,1;for L=V[2] to V[1] by -1;plt M[1]-.38,L-.14,1;lbl "10"
58: plt M[1]-.23,L-.06,1;fxd 0;lbl str(L)
59: if L≠V[1] and L≠V[2];plt M[1],L,1;plt M[1]+.08,L,2
60: next L;csiz 1.5,2,1,90;plt M[1]-.467,V[1]+1.8,1
61: lbl "Allan Variance,  Sigma (tau)";csiz .7,2,1
62: for L=M[1] to M[2] by 1;if L=M[1] or L=M[2];jmp 2
63: plt L,V[1]+.12,1;plt L,V[1],2
64: plt L-.167,V[1]-.3,1;lbl "10";plt L-.083,V[1]-.168,1;lbl str(L);next L
65: csiz 1.5,2,1;plt M[1]+1.533,V[1]-.54,1;lbl "Sample Time (tau) in seconds"
66:
67: "CALCULATE SIGMA VALUES, ERROR BARS, AND PLOT"
68: spc 3;prt " Allan Variance","  ","  Sigma     Tau","========== ====="
69: 1}M[1];2}M[2];4}M[3];8}M[4];16}M[5];32}M[6];64}M[7];128}M[8]
70: for R=8 to 1 by -1;if int(F/4M[R])<=0;next R
71: 0}A;for K=1 to 4M[R];A+(S[K+1]-S[K])^2}A;next K
72: \(.5A/(4M[R]-1)))C;fmt 9,e9.3,x,f6.0;wrt 16.9,C,I
73: \(12M[R]-4)/(4M[R]-1)}E
74: plt log(I)-.008,log(C+EC),1;plt log(I)+.008,log(C+EC),2
75: plt log(I),log(C+EC),1;plt log(I),log(C-EC),2;plt log(I)-.008,log(C-EC),1
76: plt log(I)+.008,log(C-EC),2;plt log(I),log(C),1
77: for L=2 to R;0}A}F[1]
78: for Z=1 to M[L];F[1]+S[Z]}F[1];next Z
79: F[1]/M[L]}F[1]
80: for K=M[L] to 4M[R]-M[L] by M[L];0}F[2]
81: for Z=1 to M[L];F[2]+S[K+Z]}F[2];next Z
82: F[2]/M[L]}F[2];A+(F[2]-F[1])^2}A;F[2]}F[1];next K
83: \(.5A/(4M[R]/M[L]-1)))C;wrt 16.9,C,IM[L]
84: \(12M[R]/M[L]-4)/(4M[R]/M[L]-1)}E
85: plt log(IM[L]),log(C),2;plt log(IM[L])-.008,log(C+EC),1
86: plt log(IM[L])+.008,log(C+EC),2;plt log(IM[L]),log(C+EC),1
87: plt log(IM[L]),log(C-EC),2;plt log(IM[L])-.008,log(C-EC),1
88: plt log(IM[L])+.008,log(C-EC),2;plt log(IM[L]),log(C),1;next L;spc 3
89: end
#30193
```

9.22 Program: .SIG05

```
0: "SUBPROGRAM: ALLAN VARIANCE PLOT"
1:
2: "*************************************************"
3: "*  Kevin Miller              *  File: .SIG05     *"
4: "*  January 14, 1988          *  Update: December 20, 1988  *"
5: "*  Naval Research Laboratory *  Output Device: 7245  *"
6: "*  Space Applications Branch *  Select Code: 705  *"
7: "*************************************************"
8:
9: "INITIALIZE PLOT AREA"
10: if (K-1)Z)>-6;jmp 2
11: spe 3;prt "Not Enough Data","Yet to do SIGMA","Plot.";spc 3;gto "RETURN"
12: D$[1,21]&"-"&D$[3,4]&"-"&D$[5,6])R$
13: fxd 0;hdcpy 1;ppc 705;pclr;scl -7.125,0,0,10.5;plt 0,0,1
14: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
15: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
16: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl R$
17: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.55,1;lbl "  Allan Variance"
18: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
19: plt -1/16,3.52,1;lbl F$
20: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl " Found"
21: plt -1/16,5.1,1;fxd 0;lbl str(Z)
22: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
23: fxd 2;lbl str(1/P*1e-6)&" MHz"
24: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
25: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
26: plt -9/16,9.25,1;lbl "Counter Gate";plt -5.5/16,9.6,1;lbl "Time"
27: plt -1/16,9.35,1;fxd 0;lbl str(G)&" msec";plt -1.375,10.4,1
28: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4;flt 9
29: 1)H[1];6)H[2];-15)V[1];-9)V[2]
30: scl V[2],1.23913V[1]-.23913V[2],H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
31:
32: "LABEL AXES"
33: for L=V[1]+1 to V[2]-1;plt L,H[2],1;plt L,H[2]-.083,2;next L
34: for L=H[2]-1 to H[1]+1 by -1;plt V[2],L,1;plt V[2]-.12,L,2;next L
35: csiz 1.2,1,90;for L=V[2] to V[1] by -1;plt L-.144,H[1]-.25,1;lbl "10"
36: plt L-.06,H[1]-.167,1;fxd 0;lbl str(L)
37: if L≠V[1] and L≠V[2];plt L,H[1],1;plt L,H[1]+.083,2
38: next L;csiz 1.5,2,1,180;plt V[1]+1.8,H[1]-.417,1
39: lbl "Allan Variance,  Sigma (tau)";csiz 1,2,1,90
40: for L=H[1] to H[2] by 1;if L=H[1] or L=H[2];jmp 2
41: plt V[1]+.12,L,1;plt V[1],L,2
42: plt V[1]-.3,L-.117,1;lbl "10";plt V[1]-.168,L-.05,1;fxd 0;lbl str(L)
43: next L;csiz 1.5,2,1,90;plt V[1]-.52,H[1]+1.9,1
44: lbl "Sample Time (tau) in seconds"
45:
46: "CALCULATE SIGMA VALUES, ERROR BARS, AND PLOT"
47: spc 3;prt "  Allan Variance","  ","  Sigma    Tau","********** ***** "
48: 1)M[1];2)M[2];4)M[3];8)M[4];16)M[5];32)M[6];64)M[7];128)M[8]
49: for X=8 to 1 by -1;if int((Z-2)/4M[X])<-0;next X
50: 0)L)M;rread 1,5,R$,R;gsb "READ NEXT POINT"
51: for L=1 to 4M[X]-1;gsb "READ NEXT POINT"
52: M+(S[2]-S[1])^2)M;next L
53: \(.5M/(4N[X]-1))})D;fmt 9,e9.3,x,f6.0;wrt 16.9,D,I
54: \(12M[X]-4)/(4M[X]-1))E
55: plt log(D+ED),log(I)-.008,1;plt log(D+ED),log(I)+.008,2
56: plt log(D+ED),log(I),1;plt log(D-ED),log(I),2;plt log(D-ED),log(I)-.008,1
57: plt log(D-ED),log(I)+.008,2;plt log(D),log(I),1
58: for Z=2 to X;0)L)M)H[1];rread 1,5,R$,R
59: for L=0 to M[X]-1;gsb "READ NEXT POINT"
60: H[1]+S[1])H[1];next L;H[1]/M[Z])H[1]
61: for U=M[Z] to 4M[X]-M[Z] by M[Z]
62: 0)H[2];for L=U to U+M[Z]-1;gsb "READ NEXT POINT"
63: H[2]+S[1])H[2];next L;H[2]/M[Z])H[2]
64: M+(H[2]-H[1])^2)M;H[2])H[1];next U
65: \(.5M/(4M[X]/M[Z]-1))})D;wrt 16.9,D,IM[Z]
66: \(12M[X]/M[Z]-4)/(4M[X]/M[Z]-1))E
67: plt log(D),log(IM[Z]),2;plt log(D+ED),log(IM[Z])-.008,1
68: plt log(D+ED),log(IM[Z])+.008,2;plt log(D+ED),log(IM[Z]),1
69: plt log(D-ED),log(IM[Z]),2;plt log(D-ED),log(IM[Z])-.008,1
70: plt log(D-ED),log(IM[Z])+.008,2;plt log(D),log(IM[Z]),1;next Z;spe 3
71: wtb "PRINTER",10,13,27,69
72: "RETURN";ret
73:
74: "READ NEXT POINT"
75: S[1])S[2];fxd 0;rread 1,L+6,R$,S
76: if R-S).4P;(S-P-R)/I)S[1]
77: if S-R).4P;(S+P-R)/I)S[1]
78: (S-R)/I)S[1];S)R;ret
*23276
```

9.23 Program: .SIG18

```
0: "SUBPROGRAM:  ALLAN VARIANCE PLOT"
1:
2: "*************************************************************"
3: "*  Kevin Miller                *  File: .SIG18               *"
4: "*  January 14, 1980            *  Update: December 20, 1980  *"
5: "*  Naval Research Laboratory   *  Output Device: 1350        *"
6: "*  Space Applications Branch   *  Select Code: 718           *"
7: "*************************************************************"
8:
9: "INITIALIZE PLOT AREA"
10: if (K-1)Z)>6;jmp 2
11: spc 3;prt "Not Enough Data","Yet to do SIGMA","Plot.";spc 3;gto "RETURN"
12: fxd 0;hdcpy 0;psc 718;pclr;scl 0,10.5,0,7.125;plt 0,0,1
13: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
14: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
15: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today"
16: plt .21,1/11,1;lbl D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6]&" "
17: plt 1.85,7.5/16,1;lbl "Program";plt 1.6,1/11,1;lbl " Allan Var"
18: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
19: plt 3.4,1/11,1;lbl F$&" "
20: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl " Found"
21: plt 4.95,1/11,1;fxd 0;lbl str(Z)
22: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.15,1/11,1
23: fxd 2;lbl str(1/P*1e-6)&" MHz "
24: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
25: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
26: plt 9.45,9/16,1;lbl "Gate";plt 9.45,5.7/16,1;lbl "Time"
27: plt 9.1,1/11,1;fxd 0;lbl str(G)&" msec ";plt 10.4,1.375,1
28: plt 10.4,7.125,2;plt 1,7.125;plt 1,1.375;plt 10.4,1.375;fit 9
29: 1)M[1];6)M[2];-15)V[1];-9)V[2]
30: scl M[1]-(M[2]-M[1])/9.4,M[2]+(M[2]-M[1])/94,1.23913V[1]-.23913V[2],V[2]
31:
32: "LABEL AXES"
33: for L=V[1]+1 to V[2]-1;plt M[2],L,1;plt M[2]-.08,L,2;next L
34: for L=M[2]-1 to M[1]+1 by -1;plt L,V[2],1;plt L,V[2]-.12,2;next L
35: csiz .45,2,1;for L=V[2] to V[1] by -1;plt M[1]-.38,L-.14,1;lbl "10"
36: plt M[1]-.23,L-.06,1;fxd 0;fxd 0;lbl str(L)
37: if L#V[1] and L#V[2];plt M[1],L,1;plt M[1]+.08,L,2
38: next L;csiz 1.5,2,1,90;plt M[1]-.467,V[1]+1.8,1
39: lbl "Allan Variance,  Sigma (tau)";csiz .58,2,1
40: for L=M[1] to M[2] by 1;if L=M[1] or L=M[2];jmp 2
41: plt L,V[1]+.12,1;plt L,V[1],2
42: plt L-.167,V[1]-.3,1;lbl "10";plt L-.083,V[1]-.168,1
43: fxd 0;lbl str(L);next L
44: csiz 1.5,2,1;plt M[1]+1.533,V[1]-.54,1;lbl "Sample Time (tau) in seconds"
45:
46: "CALCULATE SIGMA VALUES, ERROR BARS, AND PLOT"
47: spc 3;prt " Allan Variance"," "," Sigma    Tau","********** ***** "
48: 1)M[1];2)M[2];4)M[3];8)M[4];16)M[5];32)M[6];64)M[7];128)M[8]
49: for X=8 to 1 by -1;if int((Z-2)/4M[X])<=0;next X
50: 0)L)M;rread 1,5,R$,R;gsb "READ NEXT POINT"
51: for L=1 to 4M[X]-1;gsb "READ NEXT POINT"
52: M+(S[2]-S[1])^2)M;next L
53: \(.5M/(4M[X]-1)))D;fmt 9,e9.3,x,f6.0;wrt 16.9,D,I
54: \(12M[X]-4)/(4M[X]-1)}E
55: plt log(I)-.008,log(D+ED),1;plt log(I)+.008,log(D+ED),2
56: plt log(I),log(D+ED),1;plt log(I),log(D-ED),2;plt log(I)-.008,log(D-ED),1
57: plt log(I)+.008,log(D-ED),2;plt log(I),log(D),1
58: for Z=2 to X;0)L)M)M[1];rread 1,5,R$,R
59: for L=0 to M[X]-1;gsb "READ NEXT POINT"
60: M[1]+S[1])M[1];next L;M[1]/M[Z])M[1]
61: for U=M[Z] to 4M[X]-M[Z] by M[Z]
62: 0)M[2];for L=U to U+M[Z]-1;gsb "READ NEXT POINT"
63: M[2]+S[1])M[2];next L;M[2]/M[Z])M[2]
64: M+(M[2]-M[1])^2)M;M[2])M[1];next U
65: \(.5N/(4M[X]/M[Z]-1)))D;wrt 16.9,D,IM[Z]
66: \(12M[X]/M[Z]-4)/(4M[X]/M[Z]-1)}E
67: plt log(IM[Z]),log(D),2;plt log(IM[Z])-.008,log(D+ED),1
68: plt log(IM[Z])+.008,log(D+ED),2;plt log(IM[Z]),log(D+ED),1
69: plt log(IM[Z]),log(D-ED),2;plt log(IM[Z])-.008,log(D-ED),1
70: plt log(IM[Z])+.008,log(D-ED),2;plt log(IM[Z]),log(D),1;next Z;spc 3
71: "RETURN";ret
72:
73: "READ NEXT POINT"
74: S[1])S[2];fxd 0;rread 1,L+6,R$,S
75: if R-S).4P;(S-P-R)/I)S[1]
76: if S-R).4P;(S+P-R)/I)S[1]
77: (S-R)/I)S[1];S)R;ret
#21105
```

9.24 Program: PL005

```
0: "PROGRAM:  VOLTAGE DATA PLOT"
1:
2: "*************************************************"
3: "*  Kevin Miller                 *  File: PL005     *"
4: "*  January 14, 1980             *  Update: July 29, 1982  *"
5: "*  Naval Research Laboratory    *  Output Device: 7245    *"
6: "*  Space Applications Branch    *  Select Code: 705       *"
7: "*************************************************"
8:
9: "GET FILE NAME AND PLOT INTERVAL"
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PRINTER",706,"PLOTTER",705,"CLOCK",9
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],V[2],B$[12],E$[12]
13: dim C[26],C$[25,10],J$[25]
14: dim C,D,E,F,G,I,J,K,L,M,N,P,Q,X,Y,Z,4)K
15: ""}F$;ent "Enter Data File Name",F$;if flg13;jmp 0
16: asgn F$,1,1,0
17: if Q=1;dsp F$&" does not exist";wait 1500;jmp -2
18: asgn F$,1,1;rread 1,1,G,M,I,D$,P,J
19: if M<3;dsp "Not Enough Data for Plot";stp
20: for L=1 to J;sread 1,C[L],C$[L],J$[L,L];next L
21: ent "Plot Data from:  MMDDHHMMSS",B$
22: if flg13;"0000000000"}B$
23: if len(B$)<10;dsp "Not in Proper Form";wait 1500;jmp -2
24: ent "Plot Data to:  MMDDHHMMSS",E$
25: if flg13;"9999999999"}E$
26: if len(E$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
27: ent "Channel to be Plotted",E
28: if E<0 or E>39;dsp "No Such Channel Number";stp
29: for L=1 to J;if C[L]=E;jmp 3
30: next L;fxd 0
31: dsp "No Data Collected at Channel",E;wait 1500;jmp -4
32:
33: "CALCULATE UNITS FOR AXES"
34: 0}C}Z}D}M}Q}H[1]}H[2]}V[1]}V[2];L}E
35: rread 1,K+1}K,T$,C;if T$<B$;jmp 0
36: 1}F;for L=1 to E;sread 1,C;next L;"TIME"}H[1];C}V[1]}V[2]
37: for L=K+1 to N;rread 1,L,T$,C;if T$>E$;jmp 5
38: F+1}F;for S=1 to E;sread 1,C;next S
39: if C<V[1];C}V[1]
40: if C>V[2];C}V[2]
41: next L
42: rread 1,L-1,T$;"TIME"}H[2]
43: if H[2]-H[1]>86399;"DAYS"}U$;86400}D;jmp 4
44: if H[2]-H[1]>3599;"HOURS"}U$;3600}D;jmp 3
45: if H[2]-H[1]>59;"MINUTES"}U$;60}D;jmp 2
46: "SECONDS"}U$;1}D
47: Dint(H[1]/D)}H[1];Dint(H[2]/D+1)}H[2]
48: len(C$[E])}X;int((11-X)/2)}Y
49: for L=X+1 to 10;" "}C$[E,L,L];next L
50: for L=X+Y to Y+1 by -1;C$[E,L-Y,L-Y]}C$[E,L,L];next L
51: for L=1 to Y;" "}C$[E,L,L];next L
52: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6]}T$;4)K
53:
54: "INITIALIZE PLOT AREA"
55: fxd 0;hdcpy 1;psc 705;scl -7.125,0,0,10.5;plt 0,0,1
56: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
57: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
58: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
59: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.82,1;lbl "Channel Plot"
60: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
61: plt -1/16,3.52,1;lbl F$
62: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
63: plt -1/16,5.1,1;lbl str(F)
64: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
65: fxd 2;lbl str(1/P#1e-6)&" MHz "
66: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
67: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
68: plt -9/16,9.25,1;lbl "   Channel  ";plt -5.5/16,9.6,1;lbl "Name"
69: plt -1/16,9.35,1;lbl C$[E];plt -1.375,10.4,1
70: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4;flt 9
71: scl V[2],1.23913V[1]-.23913V[2],H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
72:
73: "LABEL AXES"
74: for L=V[1]+((V[2]-V[1])/10)C) to V[2]-C by C;plt L,H[2],1
75: plt L,H[2]-(H[2]-H[1])/100,2;next L
76: for L=H[2]-D to H[1]+D by -D;plt V[2],L,1;plt V[2]-C/7,L,2;next L
77: csiz 1,2,1,90;for L=V[2] to V[1] by -C;plt L-C/10,H[1]-(H[2]-H[1])/21,1
78: fxd 3;lbl str(L)
79: if L#V[1] or L#V[2];plt L,H[1],1;plt L,H[1]+(H[2]-H[1])/100,2
80: next L;csiz 1.5,2,1,180;plt V[1]+3.5C,H[1]-(H[2]-H[1])/12,1
81: if J$[E,E]="D";"DC Volts"}M$
82: if J$[E,E]="A";"AC Volts"}M$
83: if J$[E,E]="O";"Ohms"}M$
84: fxd 0;lbl "Channel "&str(C[E])&"  ("&M$&") ";csiz 1,2,1,90
85: for L=H[1] to H[2] by D;if L<V[1] or L>V[2];jmp 2
86: plt V[1]+C/7,L,1;plt V[1],L,2;plt V[1]-C/2.4,L-(H[2]-H[1])/100,1
```

```
87: L)Q;for M=1 to 12
88: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);;mp 3
89: if Q-2764800<0;gto "PRINT"
90: Q-2678400)Q
91: if not (M=4 or M=6 or M=9 or M=11);;mp 3
92: if Q-2678400<0;gto "PRINT"
93: Q-2592000)Q
94: if not M=2;;mp 3
95: if Q-2505600<0;gto "PRINT"
96: Q-2419200)Q
97: next M
98: "PRINT":if D=86400;fxd 0;lbl str(Q/D)
99: if D=3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
100: if D=60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
101: if D=1;fxd 0;lbl str((Q-60int(Q/60))/D)
102: next L;csiz 1.5,2,1,90;plt V[1]-.81C,H[1]+(H[2]-H[1])/3,1
103: if B$="0000000000";;mp 2
104: B$[1,2]&"-"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M$;;mp 2
105: rread 1,5,T$;T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
106: lbl "Time in "&U$&" beginning  "&M$
107:
108: "PLOT DATA":
109: rread 1,K+1)K,T$,C;if T$<B$;;mp 0
110: for L=1 to E;sread 1,C;next L
111: plt C,"TIME",1;cplt -.33,-.15;lbl "0";cplt -.67,.15
112: for L=K+1 to N;rread 1,L,T$,C;if T$>E$;;mp 3
113: for S=1 to E;sread 1,C;next S
114: plt C,"TIME",2;next L
115: pen;cplt -.33,-.15;lbl "0";wtb "PRINTER",12,13,27,69
116: end
117:
118: "TIME":
119: 0)Q;for M=1 to val(T$[1,2])-1
120: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;;mp 3
121: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;;mp 2
122: if M=2;Q+2419200)Q
123: next M
124: Q+86400val(T$[3,4])+3600val(T$[5,6]))Q
125: ret Q+60val(T$[7,8])+val(T$[9,10])
*12906
```

9.25 Program: PL018

```
0: "PROGRAM:  VOLTAGE DATA PLOT";
1:
2: "*************************************************";
3: "*  Kevin Miller                 *  File: PL018     *";
4: "*  January 14, 1980             *  Update: December 20, 1980  *";
5: "*  Naval Research Laboratory    *  Output Device: 1350        *";
6: "*  Space Applications Branch    *  Select Code: 718           *";
7: "*************************************************";
8:
9: "GET FILE NAME AND PLOT INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "CLOCK",9
12: dim F$[6],D$[6],C[26],C$[25,10],J$[25],H[2],V[2],U$[7],M$[40],T$[14]
13: dim R$[14],B$[12],E$[12],F,K,4)K
14: ""]F$;ent "Enter Data File Name",F$;if flg13;jmp 0
15: assgn F$,1,1,0;if 0#1;gto -2
16: dsp F$&" does not exist";wait 1500;gto -2
17: assgn F$,1,1;rread 1,1,G,N,I,D$,P,J
18: if J=0;spc 3;prt "No VOLTAGE DATA","Obtained this","Test.";spc 3;end
19: if N<3;spc 3;prt "Not Enough Data","to do VOLTAGE","Plot.";spc 3;end
20: ent "Plot Data from:  MMDDHHMMSS",B$
21: if flg13;"0000000000"]B$
22: if len(B$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
23: ent "Plot Data to:  MMDDHHMMSS",E$
24: if flg13;"9999999999"]E$
25: if len(E$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
26: for L=1 to J;sread 1,C[L],C$[L],J$[L,L];next L
27: ent "Channel to be Plotted",E
28: if E<0 or E>39;dsp "No Such Channel Number";stp
29: for L=1 to J;if C[L]=E;jmp 3
30: next L
31: fxd 0;dsp "No Data Collected at Channel",E;wait 1500;gto -4
32:
33: "CALCULATE UNITS FOR AXES";
34: 0)C)Z)D)M)Q)H[1])H[2])V[1])V[2];L)E
35: rread 1,K+1)K,T$,C;if T$<B$;jmp 0
36: 1)F;for S=1 to E;sread 1,C;next S
37: "DATE")H[1],C)V[1])V[2]
38: for L=K+1 to N)Z;rread 1,L,T$,C;if T$>E$;jmp 5
39: for S=1 to E;sread 1,C;next S
40: if C<V[1];C)V[1]
41: if C>V[2];C)V[2]
42: F+1)F;next L
43: rread 1,L-1,T$;"DATE")H[2]
44: if H[2]-H[1]>86399;"DAYS"]U$;86400)D;jmp 4
45: if H[2]-H[1]>3599;"HOURS"]U$;3600)D;jmp 3
46: if H[2]-H[1]>59;"MINUTES"]U$;60)D;jmp 2
47: "SECONDS"]U$;1)D
48: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2]
49: 1)W;if ((H[2]-H[1])/D)R)>30;2)W;if int(R/2)#R/2;H[2]+D)H[2]
50:
51: "INITIALIZE PLOT AREA";
52: wrt "CLOCK","R";red "CLOCK",T$;4)K
53: fxd 0;hdcpy 0;pse 718;pclr;sel 0,10.5,0,7.125;plt 0,0,1
54: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
55: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
56: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today"
57: plt .21,1/11,1;lbl T$[1,2]&"-"&T$[4,5]&"-80 "
58: plt 1.85,7.5/16,1;lbl "Program";plt 1.61,1/11,1;lbl "Channel Plt"
59: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
60: plt 3.4,1/11,1;lbl F$&" "
61: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl "Plotted"
62: plt 4.95,1/11,1;lbl str(F)&" "
63: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.15,1/11,1
64: fxd 2;lbl str(1/P#1e-6)&" MHz "
65: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
66: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
67: plt 9.28,9/16,1;lbl "Channel";plt 9.45,5.7/16,1;lbl "Name"
68: C$[E])R$;for L=1 to 5-int(len(R$)/2);" "&R$)R$;next L
69: plt 9.05,1/11,1;fxd 0;lbl R$;plt 10.4,1.375,1
70: plt 10.4,7.125,2;plt 1,7.125;plt 1,1.375;plt 10.4,1.375;flt 9
71: sel H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94,1.23913V[1]-.23913V[2],V[2]
72:
73: "LABEL AXES";
74: for L=V[1]+((V[2]-V[1])/10)C) to V[2]-C by C;plt H[2];L,1.
75: plt H[2]-(H[2]-H[1])/100,L,2;next L
76: for L=H[2]-WD to H[1]+WD by -WD;plt L,V[2],1;plt L,V[2]-C/7,2;next L
77: csiz .7,2,1;for L=V[2] to V[1] by -C;plt H[1]-(H[2]-H[1])/12.5,L-C/10,1
78: fxd 2;lbl str(L)
79: if L#V[1] and L#V[2];plt H[1],L,1;plt H[1]+(H[2]-H[1])/100,L,2
80: next L;csiz 1.5,2,1,90;plt H[1]-(H[2]-H[1])/11,V[1]+3.1C,1
81: if J$[E,E]="D";"DC Volts")M$
82: if J$[E,E]="A";"AC Volts")M$
83: if J$[E,E]="O";"Ohms")M$
84: fxd 0;lbl "Channel "&str(C[E])&"   ("&M$&") ";csiz .7,2,1
85: for L=H[1] to H[2] by WD;if L=H[1] or L=H[2];jmp 2
86: plt L,V[1]+C/7,1;plt L,V[1],2
```

```
87: plt L-(H[2]-H[1])/50,V[1]-.35C,1;L)0;for M=1 to 12
88: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);;jmp 3
89: if Q-2764800<0;qto "PRINT"
90: Q-2678400)@
91: if not (M=4 or M=6 or M=9 or M=11);;jmp 3
92: if Q-2678400<0;qto "PRINT"
93: Q-2592000)@
94: if not M=2;;jmp 3
95: if Q-2505600<0;qto "PRINT"
96: Q-2419200)@
97: next M
98: "PRINT";if D=86400;fxd 0;lbl str(Q/D)
99: if D=3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
100: if D=60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
101: if D=1;fxd 0;lbl str((Q-60int(Q/60))/D)
102: next L;csiz 1.5,2,1;plt H[1]+(H[2]-H[1])/4.2,V[1]-.85C,1
103: if B$="0000000000";;jmp 2
104: B$[1,2]&"-"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10]]M$;;jmp 2
105: rread 1,5,T$;T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10]]M$
106: lbl "Time in "&U$&" beginning  "&M$&" "
107:
108: "PLOT DATA";
109: rread 1,K+1)K,T$,C;if T$<B$;;jmp 0
110: for S=1 to E;sread 1,C;next S
111: plt "DATE",C,1;cplt -.33,-.15;lbl "o";cplt -.67,.15
112: for L=K+1 to N;rread 1,L,T$,C;if T$>E$;;jmp 3
113: for S=1 to E;sread 1,C;next S
114: plt "DATE",C,2;next L
115: pen;cplt -.33,-.15;lbl "o"
116: end
117:
118: "DATE";
119: 0)Q;for M=1 to val(T$[1,2])-1
120: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;;jmp 3
121: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;;jmp 2
122: if M=2;Q+2419200)@
123: next M
124: Q+86400val(T$[3,4])+3600val(T$[5,6]))@
125: ret Q+60val(T$[7,8])+val(T$[9,10])
#32235
```

9.26 Program: .PL005

```
0: "SUBPROGRAM,  VOLTAGE DATA PLOT";
1:
2: "*************************************************";
3: "*  Kevin Miller                 *  File: .PL005    *";
4: "*  January 14, 1980             *  Update: December 28, 1980  *";
5: "*  Naval Research Laboratory    *  Output Device: 7245        *";
6: "*  Space Applications Branch    *  Select Code: 705           *";
7: "*************************************************";
8:
9: "DETERMINE WHICH CHANNEL TO PLOT";
10: if J>0;jmp 3
11: spc 3;prt "No VOLTAGE DATA","will be Obtained"
12: prt "this Test.";spc 3;gto "RETURN"
13: if K>2;jmp 3
14: spc 3;prt "Not Enough Data","Yet to do","VOLTAGE CHANNEL"
15: prt "Plot.";spc 3;gto "RETURN"
16: spc 3;prt "----------------","  VOLTAGE PLOT","----------------";spc 2
17: prt "Keyin Special"," Function Key:";spc ;prt "f0 to increment"
18: prt "f1 to decrement","f2 to plot";spc 3;1)E
19: "CYCLE",fxd 0;str(C(E))U$;if C(E)<10;" 0"&U$(2,2)U$
20: dsp "Channel Number:"&U$&"  ["&C$(E)&"]"
21: for L=0 to 2;if flg(L;cfg L;jmp 2L+2
22: next L;gto "CYCLE"
23: "f0";if E=J;gto "CYCLE"
24: "f0";E+1)E;gto "CYCLE"
25: "f1";if E=1;gto "CYCLE"
26: "f1";E-1)E;gto "CYCLE"
27: "f2";dsp "Plot Channel:"&U$&"   ["&C$(E)&"]"
28:
29: "CALCULATE UNITS FOR AXES";
30: rread 1,5,R$,R;for S=1 to E;sread 1,R;next S
31: "DATE")H(1),R)V(1))V(2)
32: for L=2 to K-1)Z;rread 1,L+4,R$,R;0)Y
33: rread 1,L+4,R$,R;for S=1 to E;sread 1,R;next S
34: if R<V(1);R)V(1)
35: if R>V(2);R)V(2)
36: next L;"DATE")H(2)
37: if H(2)-H(1)>86399;"DAYS")U$;86400)D;jmp 4
38: if H(2)-H(1)>3599;"HOURS")U$;3600)D;jmp 3
39: if H(2)-H(1)>59;"MINUTES")U$;60)D;jmp 2
40: "SECONDS")U$;1)0
41: Dint(H(1)/D))H(1);Dint(H(2)/D+1))H(2)
42: D$(1,2)&"-"&D$(3,4)&"-"&D$(5,6))R$
43:
44: "INITIALIZE PLOT AREA";
45: fxd 0;hdcpy 1;psc 705;plr;sel -7.125,0,0,10.5;plt 0,0,1
46: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
47: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
48: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl R$
49: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.82,1;lbl "Channel Plot"
50: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
51: plt -1/16,3.52,1;lbl F$
52: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
53: plt -1/16,5.1,1;fxd 0;lbl str(Z)
54: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
55: fxd 2;lbl str(1/P*1e-6)&" MHz "
56: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
57: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
58: C$(E))R$;for L=1 to 5-int(len(R$)/2);" "&R$)R$;next L
59: plt -9/16,9.25,1;lbl "   Channel";plt -5.5/16,9.6,1;lbl "Name"
60: plt -1/16,9.35,1;lbl R$;plt -1.375,10.4,1
61: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4;flt 9
62: sel V(2),1.23913V(1)-.23913V(2),H(1)-(H(2)-H(1))/9.4,H(2)+(H(2)-H(1))/94
63:
64: "LABEL AXES";
65: for L=V(1)+((V(2)-V(1))/10)R) to V(2)-R by R;plt L,H(2),1
66: plt L,H(2)-(H(2)-H(1))/100,2;next L
67: for L=H(2)-D to H(1)+D by -D;plt V(2),L,1;plt V(2)-R/7,L,2;next L
68: csiz 1,2,1,90;for L=V(2) to V(1) by -R;plt L-R/10,H(1)-(H(2)-H(1))/21,1
69: fxd 3;lbl str(L)
70: if L=V(1) or L=V(2);plt L,H(1),1;plt L,H(1)+(H(2)-H(1))/100,2
71: next L;csiz 1.5,2,1,180;plt V(1)+3.51R,H(1)-(H(2)-H(1))/12,1
72: if J$(E,E)="D";"DC Volts")M$
73: if J$(E,E)="A";"AC Volts")M$
74: if J$(E,E)="O";"Ohms")M$
75: fxd 0;lbl "Channel "&str(C(E))&"  ("&M$&") ";csiz 1,2,1,90
76: for L=H(1) to H(2) by D;if L=V(1) or L=V(2);jmp 2
77: plt V(1)+R/7,L,1;plt V(1),L,2;plt V(1)-R/2.4,L-(H(2)-H(1))/100,1
78: L)X;for M=1 to 12
79: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
80: if X-2764800<0;gto "PRINT"
81: X-2678400)X
82: if not (M=4 or M=6 or M=9 or M=11);jmp 3
83: if X-2678400<0;gto "PRINT"
84: X-2592000)X
85: if not M=2;jmp 3
86: if X-2505600<0;gto "PRINT"
```

```
87: X-2419200)X
88: next M
89: "PRINT";if D-86400;fxd 0;lbl str(X/D)
90: if D-3600;fxd 0;lbl str((X-86400int(X/86400))/D)
91: if D-60;fxd 0;lbl str((X-3600int(X/3600))/D)
92: if D-1;fxd 0;lbl str((X-60int(X/60))/D)
93: next L;csiz 1.5,2.1,90;plt V(1)-.81R,H(1)+(H(2)-H(1))/3,1
94: rread 1,5,R$;R$[1,2]&":"&R$[3,4]&"  "&R$[5,6]&":"&R$[7,8]&":"&R$[9,10])M$
95: lbl "Time in "&U$&" beginning  "&M$
96:
97: "PLOT DATA":
98: rread 1,5,R$,R;for S=1 to E;sread 1,R;next S
99: plt R,"DATE",1;cplt -.33,-.15;lbl "e";cplt -.67,.15
100: for L=2 to Z
101: rread 1,L*4,R$,R;for S=1 to E;sread 1,R;V*1)Y;next S
102: plt R,"DATE",2;next L
103: pen;cplt -.33,-.15;lbl "e"
104: wtb "PRINTER",10,13,27,69
105: "RETURN";ret
106:
107: "DATE":
108: 0)X;for M=1 to val(R$[1,2])-1
109: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;X+2678400)X;)mp 3
110: if M=4 or M=6 or M=9 or M=11;X+2592000)X;)mp 2
111: if M=2;X+2419200)X
112: next M
113: X+86400val(R$[3,4])+3600val(R$[5,6])X
114: ret X+60val(R$[7,8])+val(R$[9,10])
#13152
```

9.27 Program: .PL018

```
0: "SUBPROGRAM:  VOLTAGE DATA PLOT";
1:
2: "*************************************************";
3: "*  Kevin Miller              *  File: .PL018      *";
4: "*  January 14, 1988          *  Update: December 20, 1988  *";
5: "*  Naval Research Laboratory *  Output Device: 1358       *";
6: "*  Space Applications Branch *  Select Code: 718          *";
7: "*************************************************";
8:
9: "DETERMINE WHICH CHANNEL TO PLOT";
10: if J>0;jmp 3
11: spc 3;prt "No VOLTAGE DATA","will be Obtained"
12: prt "this Test.";spc 3;gto "RETURN"
13: if K>2;jmp 3
14: spc 3;prt "Not Enough Data","Yet to do","VOLTAGE CHANNEL"
15: prt "Plot.";spc 3;gto "RETURN"
16: spc 3;prt "---------------","  VOLTAGE PLOT","---------------";spc 2
17: prt "Kevin Special","  Function Keys:";spc ;prt "f0 to increment"
18: prt "f1 to decrement","f2 to plot";spc 3;1)E
19: "CYCLE";fxd 0;str(C[E])U$;if C[E]<10;" 0"&U$[2,2])U$
20: dsp "Channel Number:"&U$&"  ["&C$[E]&"]"
21: for L=0 to 2;if flg[L;cfg L;jmp 2L+2
22: next L;gto "CYCLE"
23: "f0";if E=J;gto "CYCLE"
24: "f0";E+1)E;gto "CYCLE"
25: "f1";if E=1;gto "CYCLE"
26: "f1";E-1)E;gto "CYCLE"
27: "f2";dsp "Plot Channel:"&U$&"   ["&C$[E]&"]"
28:
29: "CALCULATE UNITS FOR AXES";
30: hdcpy 0;psc 718;pclr
31: rread 1,5,R$,R;for S=1 to E;sread 1,R;next S
32: "DATE")H[1];R)V[1])V[2]
33: for L=2 to K-1)Z
34: rread 1,L*4,R$,R;for S=1 to E;sread 1,R;next S
35: if R<V[1];R)V[1]
36: if R>V[2];R)V[2]
37: next L;"DATE")H[2]
38: if H[2]-H[1]>86399;"DAYS")U$;86400)D;jmp 4
39: if H[2]-H[1]>3599;"HOURS")U$;3600)D;jmp 3
40: if H[2]-H[1]>59;"MINUTES")U$;60)D;jmp 2
41: "SECONDS")U$;1)D
42: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2]
43: 1)W;if ((H[2]-H[1])/D)R)>30;2)W;if int(R/2)#R/2;H[2]+D)H[2]
44:
45: "INITIALIZE PLOT AREA";
46: fxd 0;scl 0,10.5,0,7.125;plt 0,0,1
47: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
48: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
49: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today "
50: plt .21,1/11,1;lbl D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6]&" "
51: plt 1.85,7.5/16,1;lbl "Program";plt 1.61,1/11,1;lbl "Channel Plt"
52: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
53: plt 3.4,1/11,1;lbl F$&" "
54: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl "Plotted"
55: plt 4.95,1/11,1;fxd 0;lbl str(Z)&" "
56: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.15,1/11,1
57: fxd 2;lbl str(1/P#1e-6)&" MHz "
58: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
59: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
60: plt 9.28,9/16,1;lbl "Channel";plt 9.45,5.7/16,1;lbl "Name"
61: C$[E])R$ for L=1 to 5-int(len(R$)/2);" "&R$)R$;next L
62: plt 9.05,1/11,1;fxd 0;lbl R$;plt 10.4,1.375,1
63: plt 10.4,7.125,2;plt 1,7.125;plt 1,1.375;plt 10.4,1.375;flt 9
64: scl H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94,1.23913V[1]-.23913V[2],V[2]
65:
66: "LABEL AXES";
67: for L=V[1]+((V[2]-V[1])/10)R) to V[2]-R by R;plt H[2],L,1
68: plt H[2]-(H[2]-H[1])/100,L,2;next L
69: for L=H[2]-WD to H[1]+WD by -WD;plt L,V[2],1;plt L,V[2]-R/7,2;next L
70: csiz .7,2,1;for L=V[2] to V[1] by -R;plt H[1]-(H[2]-H[1])/12.5,L-R/10,1
71: fxd 2;lbl str(L)
72: if L#V[1] and L#V[2];plt H[1],L,1;plt H[1]+(H[2]-H[1])/100,L,2
73: next L;csiz 1.5,2,1,90;plt H[1]-(H[2]-H[1])/11,V[1]+3.1R,1
74: if J$[E,E]="D";"DC Volts")M$
75: if J$[E,E]="A";"AC Volts")M$
76: if J$[E,E]="O";"Ohms")M$
77: fxd 0;lbl "Channel "&str(C[E])&"   ("&M$&") ";csiz .7,2,1
78: for L=H[1] to H[2] by WD;if L=H[1] or L=H[2];jmp 2
79: plt L,V[1]+R/7,1;plt L,V[1],2
80: plt L-(H[2]-H[1])/50,V[1]-.35R,1;L)X;for M=1 to 12
81: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
82: if X-2764800<0;gto "PRINT"
83: X-2678400)X
84: if not (M=4 or M=6 or M=9 or M=11);jmp 3
85: if X-2678400<0;gto "PRINT"
86: X-2592000)X
```

```
87: if not M-2;jmp 3
88: if X-2505600<0;gto "PRINT"
89: X-2419200)X
90: next M
91: "PRINT";if D-86400;fxd 0;lbl str(X/D)
92: if D-3600;fxd 0;lbl str((X-86400int(X/86400))/D)
93: if D-60;fxd 0;lbl str((X-3600int(X/3600))/D)
94: if D-1;fxd 0;lbl str((X-60int(X/60))/D)
95: next L;csiz 1.5,2,1;plt H(1)+(H(2)-H(1))/4.2,V(1)-.85R,1
96: rread 1,5,R$;R$[1,2]&"-"&R$[3,4]&"  "&R$[5,6]&":"&R$[7,8]&":"&R$[9,10])M$
97: lbl "Time in "&U$&" beginning  "&M$&" "
98:
99: "PLOT DATA";
100: rread 1,5,R$,R;for S=1 to E;sread 1,R;next S
101: plt "DATE",R,1;cplt -.33,-.15;lbl "o";cplt -.67,.15
102: for L=2 to Z
103: rread 1,L+4,R$,R;for S=1 to E;sread 1,R;next S
104: plt "DATE",R,2;next L
105: pen;cplt -.33,-.15;lbl "o"
106: "RETURN";ret
107:
108: "DATE";
109: 0)X;for M=1 to val(R$[1,2])-1
110: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;X+2678400)X;;jmp 3
111: if M=4 or M=6 or M=9 or M=11;X+2592000)X;;jmp 2
112: if M=2;X+2419200)X
113: next M
114: X+86400val(R$[3,4])+3600val(R$[5,6])X
115: ret X+60val(R$[7,8])+val(R$[9,10])
#9252
```

9.28 Program: MLT05

```
0: "PROGRAM:  ANALOG DATA MULTIPLOT";
1:
2: "*************************************************************";
3: "*  Kevin Miller                 *  File: MLT05                 *";
4: "*  August 2, 1982               *  Update: August 2, 1982      *";
5: "*  Naval Research Laboratory    *  Output Device: 7245         *";
6: "*  Space Applications Branch    *  Select Code: 705            *";
7: "*************************************************************";
8:
9: "GET FILE NAME AND PLOT INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PRINTER",706,"PLOTTER",705,"CLOCK",9
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],V[4,2],B$[12],E$[12]
13: dim A[4],A$[4,3],P$[4,1],C[26],C$[25,10],J$[25]
14: dim C,D,E,F,G,I,J,K,L,M,N,P,Q,X,Y,Z,4)K
15: "1st")A$[1];"2nd")A$[2];"3rd")A$[3];"4th")A$[4]
16: "o")P$[1];"x")P$[2];"/")P$[3];"$")P$[4]
17: "")F$;ent "Enter Data File Name",F$;if flg13;jmp 0
18: asgn F$,1,1,0
19: if 0=1;dsp F$&" does not exist";wait 1500;jmp -2
20: asgn F$,1,1;rread 1,1,G,M,I,D$,P,J
21: if N<3;dsp "Not Enough Data for Plot";stp
22: for L=1 to J;sread 1,C[L],C$[L],J$[L,L];next L
23: ent "Plot Data from:  MMDDHHMMSS",B$
24: if flg13;"0000000000")B$
25: if len(B$)<10;dsp "Not in Proper Form";wait 1500;jmp -2
26: ent "Plot Data to:  MMDDHHMMSS",E$
27: if flg13;"9999999999")E$
28: if len(E$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
29: for L=1 to 4
30: dsp "Enter "&A$[L]&" Channel to Plot"
31: ent "",A[L];if not flg13;jmp 3
32: if L<3;jmp -2
33: L-1)B;jmp 5
34: if A[L]<0 or A[L]>39;dsp "No Such Channel Number";wait 1500;jmp -2
35: for M=1 to J;if C[M]#A[L];next M;jmp 2
36: M)A[L];next L;4)B;jmp 2
37: fxd 0;dsp "No Data Collected at Channel"&str(A[L]);wait 1500;jmp -7
38: for L=1 to B-1;for M=L+1 to B;if A[M]<A[L];jmp 2
39: A[L])A;A[M])A[L];A)A[M]
40: next M;next L
41:
42: "CALCULATE UNITS FOR AXES";
43: rread 1,K+1)K,T$,C;if T$<B$;jmp 8
44: K)Y;1)F;0)M;for L=1 to B;for M=1 to A[L]-M;sread 1,C;next M
45: "TIME")H[1];C)V[L,1])V[L,2];A[L])M;next L
46: for L=K+1 to M;rread 1,L,T$,C;if T$>E$;jmp 5
47: F+1)F;0)M;for M=1 to B;for O=1 to A[M]-M;sread 1,C;next O
48: if C<V[M,1];C)V[M,1]
49: if C>V[M,2];C)V[M,2]
50: A[M])M;next M;next L
51: rread 1,L-1)Z,T$;"TIME")H[2]
52: if H[2]-H[1]>86399;"DAYS")U$;86400)D;jmp 4
53: if H[2]-H[1]>3599;"HOURS")U$;3600)D;jmp 3
54: if H[2]-H[1]>59;"MINUTES")U$;60)D;jmp 2
55: "SECONDS")U$;1)D
56: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2];H[2]-H[1])M
57: for L=1 to J;for M=1 to int(5-len(C$[L])/2)
58: " "&C$[L])C$[L];next M;next L
59: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])T$;4)K
60:
61: "INITIALIZE PLOT AREA";
62: hdcpy 1;fxd 0;scl -7.125,0,0,10.5;plt 0,0,1
63: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
64: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
65: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
66: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.62,1;lbl "Analog Multiplot"
67: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
68: plt -1/16,3.52,1;lbl F$
69: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
70: plt -1/16,5.1,1;lbl str(F)
71: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
72: fxd 2;lbl str(1/P#1e-6)&" MHz "
73: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
74: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
75: plt -9/16,9.3,1;lbl "  Channel";plt -5.5/16,9.4,1;lbl "Function"
76: "DC Voltage")M$;if J$[A[1],A[1]]="A";"A")M$[1,1]
77: if J$[A[1],A[1]]="O";"   Ohms")M$[1,1]
78: plt -1/16,9.3,1;lbl M$;plt -1.375,10.4,1
79: plt -7.125,10.4,2;plt -7.125,8;plt -1.375,8;plt -1.375,10.4;flt 9
80: H[2]-10.4W/(10.4-8))R
81: scl V[1,2],1.23913V[1,1]-.23913V[1,2],R,(10.5H[2]-.1R)/10.4
82:
83: "LABEL AXES";
84: for L=V[1,1]+((V[1,2]-V[1,1])/10)C) to V[1,2]-C by C;plt L,H[2],1
85: plt L,H[2]-.104W/(10.4-8),2;next L
86: for L=H[2]-D to H[1]+D by -D;plt V[1,2],L,1;plt V[1,2]-C/7,L,2;next L
```

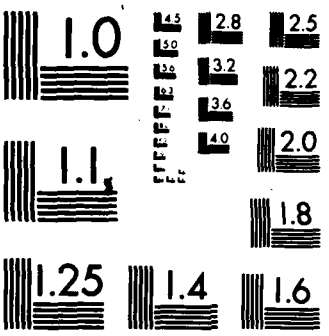
```
87: for L=V[1,2] to V[1,1) by -C;plt L-C/10,H[1)-.42W/(10.4-B),1
88: csiz 1,2,1,90;fxd 3;lbl str(L)
89: if L<V[1,1] and L>V[1,2];plt L,H[1),1;plt L,H[1)+.104W/(10.4-B),2
90: next L;csiz 1.5,2,1,180;plt V[1,1)+3.2C,H[1)-.55W/(10.4-B),1
91: fxd 0;lbl P$[1]&"   "&C$[A[1]]&"   (Channel"&str(C[A[1]])&")"
92: for D=2 to B;csiz 1,2,1,90
93: scl V[0,2],1.23913V[0,1)-.23913V[0,2],R,(10.5H[2)-.1R)/10.4
94: plt V[0,1],H[1)-W(D-1)/(10.4-B),1;plt V[0,2),H[1)-W(D-1)/(10.4-B),2
95: for L=V[0,2] to V[0,1] by -((V[0,2]-V[0,1))/10)C)
96: plt L-C/10,H[1)-W(D-.58)/(10.4-B),1;fxd 3;lbl str(L)
97: plt L,H[1)-W(D-1)/(10.4-B),1;plt L,H[1]-W(D-1.104)/(10.4-B),2;next L
98: csiz 1.5,2,1,180;plt V[0,1)+3.2C,H[1)-W(D-.45)/(10.4-B),1
99: fxd 0;lbl P$[0]&"   "&C$[A[0]]&"   (Channel"&str(C[A[0]])&")";next D
100: csiz 1,2,1,90;D-1)D;for L=H[1] to H[2] by D;if L=H[1] or L=H[2];jmp 2
101: plt V[0,1)+C/7,L,1;plt V[0,1],L,2
102: plt V[0,1]-C/2.4,L-.104W/(10.4-B),1;L)Q;for M=1 to 12
103: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
104: if Q-2764800<0;gto "PRINT"
105: Q-2678400)Q
106: if not (M=4 or M=6 or M=9 or M=11);jmp 3
107: if Q-2678400<0;gto "PRINT"
108: Q-2592000)Q
109: if not M-2;jmp 3
110: if Q-2505600<0;gto "PRINT"
111: Q-2419200)Q
112: next M
113: "PRINT";if D-86400;fxd 0;lbl str(Q/D)
114: if D-3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
115: if D-60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
116: if D-1;fxd 0;lbl str((Q-60int(Q/60))/D)
117: next L;csiz 1.5,2,1,90;plt V[0,1]-.81C,H[1]+(7.2-B)W/(20.8-2B),1
118: if B$-"0000000000";jmp 2
119: B$[1,2]&"-"&B$[3,4]&"   "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M$;jmp 2
120: rread 1,5,T$;T$[1,2]&"-"&T$[3,4]&"   "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
121: lbl "Time in "&U$&" beginning   "&M$
122:
123: "PLOT DATA":
124: for D=1 to B
125: scl V[0,2],1.23913V[0,1]-.23913V[0,2],R,(10.5H[2]-.1R)/10.4
126: rread 1,Y,T$,C
127: for M=1 to A[0];sread 1,C;next M
128: plt C,"TIME",1;cplt -.33,-.15;lbl P$[0];plt C,"TIME",1
129: for L=Y+1 to Z;rread 1,L,T$,C
130: for M=1 to A[0];sread 1,C;next M
131: plt C,"TIME",2;next L
132: cplt -.33,-.15;lbl P$[0];next D
133: wtb "PRINTER",12,13,27,69
134: end
135:
136: "TIME":
137: 0)Q;for M=1 to val(T$[1,2])-1
138: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;jmp 3
139: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;jmp 2
140: if M=2;Q+2419200)Q
141: next M
142: Q+86400val(T$[3,4])+3600val(T$[5,
143: ret Q+60val(T$[7,8])+val(T$[9,10
■3703
```

9.29 Program: MLT13

```
0: "PROGRAM:  ANALOG DATA MULTIPLOT DISPLAY";
1:
2: "*************************************************";
3: "*  Kevin Miller,Trip Carter   *  File: MLT18      *";
4: "*  July 20, 1983              *  Update: July 20, 1983  *";
5: "*  Naval Research Laboratory  *  Output Device: 7245  *";
6: "*  Space Applications Branch  *  Select Code: 718    *";
7: "*************************************************";
8:
9: "GET FILE NAME AND PLOT INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "SCREEN",718,"CLOCK",9
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],V[4,2],B$[12],E$[12]
13: dim A[4],A$[4,3],P$[4,1],C[26],C$[25,10],J$[25]
14: dim C,D,E,F,G,I,J,K,L,M,N,P,Q,X,Y,Z,4)K
15: "1st")A$[1];"2nd")A$[2];"3rd")A$[3];"4th")A$[4]
16: "o")P$[1];"x")P$[2];"/")P$[3];"$")P$[4]
17: "")F$;ent "Enter Data File Name",F$;if flg13;jmp 0
18: asgn F$,1,1,0
19: if Q=1;dsp F$&" does not exist";wait 1500;;jmp -2
20: asgn F$,1,1;rread 1,1,G,M,I,D$,P,J
21: if N<3;dsp "Not Enough Data for Plot";stp
22: for L=1 to J;sread 1,C[L],C$[L],J$[L,L];next L
23: ent "Plot Data from:  MMDDHHMMSS",B$
24: if flg13;"0000000000")B$
25: if len(B$)<10;dsp "Not in Proper Form";wait 1500;;jmp -2
26: ent "Plot Data to:  MMDDHHMMSS",E$
27: if flg13;"9999999999")E$
28: if len(E$)#10;dsp "Not in Proper Form";wait 1500;;jmp -2
29: for L=1 to 4
30: dsp "Enter "&A$[L]&" Channel to Plot"
31: ent "",A[L];if not flg13;;jmp 3
32: if L<3;;jmp -2
33: L-1)B;;jmp 5
34: if A[L]<0 or A[L]>39;dsp "No Such Channel Number";wait 1500;;jmp -2
35: for M=1 to J;if C[M]#A[L];next M;;jmp 2
36: M)A[L];next L;4)B;;jmp 2
37: fxd 0;dsp "No Data Collected at Channel"&str(A[L]);wait 1500;;jmp -7
38: for L=1 to B-1;for M=L+1 to B;if A[M]>A[L];;jmp 2
39: A[L])A;A[M])A[L];A)A[M]
40: next M;next L
41:
42: "CALCULATE UNITS FOR AXES";
43: rread 1,K+1)K,T$,C;if T$<B$;;jmp 0
44: K)Y;1)F;0)H;for L=1 to B;for M=1 to A[L]-H;sread 1,C;next M
45: "TIME")H[1];C)V[L,1])V[L,2];A[L])H;next L
46: for L=K+1 to N;rread 1,L,T$,C;if T$>E$;;jmp 5
47: F+1)F;0)H;for M=1 to B;for O=1 to A[M]-H;sread 1,C;next O
48: if C<V[M,1];C)V[M,1]
49: if C>V[M,2];C)V[M,2]
50: A[M])H;next M;next L
51: rread 1,L-1)Z,T$;"TIME")H[2]
52: if H[2]-H[1]>86399;"DAYS")U$;86400)D;;jmp 4
53: if H[2]-H[1]>3599;"HOURS")U$;3600)D;;jmp 3
54: if H[2]-H[1]>59;"MINUTES")U$;60)D;;jmp 2
55: "SECONDS")U$;1)D
56: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2];H[2]-H[1])M
57: for L=1 to J;for M=1 to int(5-len(C$[L])/2)
58: " "&C$[L])C$[L];next M;next L
59: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])T$;"K
60:
61: "INITIALIZE PLOT AREA";
62: fxd 0;hdcpy 0;psc 718;pclr;scl 0,10.5,0,7.125;plt 0,0,1
63: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
64: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
65: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today"
66: plt .21,1/11,1;lbl T$[1,2]&"-"&T$[4,5]&"-80 "
67: plt 1.85,7.5/16,1;lbl "Program";plt 1.61,1/11,1;lbl " Multiplot "
68: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
69: plt 3.4,1/11,1;lbl F$&" "
70: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl "Plotted"
71: plt 4.95,1/11,1;lbl str(F)&" "
72: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.15,1/11,1
73: fxd 2;lbl str(1/P*1e-6)&" MHz "
74: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
75: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
76: plt 9.18,9/16,1;lbl "Channel";plt 9.18,5.7/16,1;lbl "Function"
77: "DC Voltage")M$;if J$[A[1],A[1]]="A";"A")M$[1,1]
78: if J$[A[1],A[1]]="O";"    Ohms")M$
79: plt 9.1,1/16,1;lbl M$;plt 10.4,1.375,1
80: plt 10.4,1.375,1;plt 10.4,7.125,2;plt 8,7.125;plt P,1.375
81: plt 10.4,1.375;H[2]-10.4W/(10.4-B))R
82: scl R,(10.5H[2]-.1R)/10.4,1.23913V[1,1]-.23913V[1,2],V[1,2]
83:
84: "LABEL AXES";
85: for L=V[1,1]+((V[1,2]-V[1,1])/10)C) to V[1,2]-C by C;plt H[2],L,1
86: plt H[2]-.104W/(10.4-B),L,2;next L
```

AD-A137 712 THERMAL VACUUM TEST FACILITY(U) NAVAL RESEARCH LAB WASHINGTON DC K M MILLER 31 JAN 84 NRL-MR-5217 2/2

UNCLASSIFIED F/G 14/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
87: for L=H[2]-0 to H[1]+0 by -D;plt L,V[1,2],1;plt L,V[1,2]-C/7.2;next L
88: for L=V[1,2] to V[1,1] by -C;plt H[1]-.68W/(10.4-8),L-C/10,1
89: csiz 1.2,1.0;fxd 2;lbl str(L)
90: if L≠V[1,1] and L≠V[1,2];plt H[1],L,1;plt H[1]+.104W/(10.4-8),L,2
91: next L;csiz 1.5,2,1.90;plt H[1]-.65W/(10.4-8),V[1,1]+3.2C,1
92: fxd 0;lbl " "&P$[1]&"   "&C$[A[1]]&"   (Channel"&str(C[A[1]])&") "
93: for D=2 to 8;csiz 1.2,1.0
94: scl R,(10.5H[2]-.1R)/10.4,1.23913V[0,1]-.23913V[0,2],V[0,2]
95: plt H[1]-W(D-1)/(10.4-8),V[0,1],1;plt H[1]-W(D-1)/(10.4-8),V[0,2],2
96: for L=V[0,2] to V[0,1] by -((V[0,2]-V[0,1])/10)C)
97: plt H[1]-W(D-.32)/(10.4-8),L-C/10,1;fxd 2;lbl str(L)
98: plt H[1]-W(D-1)/(10.4-8),L,1;plt H[1]-W(D-1.104)/(10.4-8),L,2;next L
99: csiz 1.5,2,1.90;plt H[1]-W(D-.33)/(10.4-8),V[0,1]+3.2C,1
100: fxd 0;lbl " "&P$[D]&"   "&C$[A[D]]&"   (Channel"&str(C[A[D]])&") ";next D
101: csiz 1.2,1.0;D-1)D;for L=H[1] to H[2] by D;if L=H[1] or L=H[2];jmp 2
102: plt L,V[0,1]+C/7,1;plt L,V[0,1],2
103: plt L-.184W/(10.4-8),V[0,1]-C/2.4,1;L)Q;for M=1 to 12
104: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
105: if Q-2764800<0;gto "PRINT"
106: Q-2678400)Q
107: if not (M=4 or M=6 or M=9 or M=11);jmp 3
108: if Q-2678400<0;gto "PRINT"
109: Q-2592000)Q
110: if not M=2;jmp 3
111: if Q-2505600<0;gto "PRINT"
112: Q-2419200)Q
113: next M
114: "PRINT":if D=86400;fxd 0;lbl str(Q/D)
115: if D=3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
116: if D=60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
117: if D=1;fxd 0;lbl str((Q-60int(Q/60))/D)
118: next L;csiz 1.5,2,1.0;plt H[1]+.46(7.2-8)W/(20.8-28),V[0,1]-.81C,1
119: if B$="0000000000";jmp 2
120: B$[1,2]&"-"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M$;jmp 2
121: rread 1,5,T$;T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
122: lbl "Time in "&U$&" beginning  "&M$&" "
123:
124: "PLOT DATA":
125: for D=1 to 8
126: scl R,(10.5H[2]-.1R)/10.4,1.23913V[0,1]-.23913V[0,2],V[0,2]
127: rread 1,Y,T$,C
128: for M=1 to A[0];sread 1,C;next M
129: plt "TIME",C,1;cplt -.15,-.33;lbl P$[0];plt "TIME",C,1
130: for L=Y+1 to Z;rread 1,L,T$,C
131: for M=1 to A[0];sread 1,C;next M
132: plt "TIME",C,2;next L
133: cplt -.15,-.33;lbl P$[0];next D
134: end
135:
136: "TIME":
137: 0)Q;for M=1 to val(T$[1,2])-1
138: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;jmp 3
139: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;jmp 2
140: if M=2;Q+2419200)Q
141: next M
142: Q+86400val(T$[3,4])+3600val(T$[5,6]))Q
143: ret Q+60val(T$[7,8])+val(T$[9,10])
#15316
```

9.30 Program: STK05

```
0: "PROGRAM:  ANALOG DATA STACK PLOT";
1:
2: "*************************************************************";
3: "*  Kevin Miller                 *  File: STK05                 *";
4: "*  August 5, 1982               *  Update: August 5, 1982      *";
5: "*  Naval Research Laboratory    *  Output Device: 7245         *";
6: "*  Space Applications Branch    *  Select Code: 705            *";
7: "*************************************************************";
8:
9: "GET FILE NAME AND PLOT INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PRINTER",706,"PLOTTER",705,"CLOCK",9
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],V[4,2],B$[12],E$[12]
13: dim A[4],A$[4,3],C[26],C$[25,10],J$[25]
14: dim C,D,E,F,G,I,J,K,L,M,N,P,Q,X,Y,Z;4)K
15: "1st")A$[1],"2nd")A$[2],"3rd")A$[3],"4th")A$[4]
16: ""}F$;ent "Enter Data File Name",F$;if flg13;jmp 0
17: assn F0,1,1,0
18: if Q=1;dsp F$&" does not exist";wait 1500;;jmp -2
19: assn F$,1,1;rread 1,1,G,M,I,D$,P,J
20: if M<3;dsp "Not Enough Data for Plot";stp
21: for L=1 to J;rread 1,C[L],C$[L],J$[L,L];next L
22: ent "Plot Data from:  MMDDHHMMSS",B$
23: if flg13;"0000000000")B$
24: if len(B$)<10;dsp "Not in Proper Form";wait 1500;jmp -2
25: ent "Plot Data to:  MMDDHHMMSS",E$
26: if flg13;"9999999999")E$
27: if len(E$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
28: for L=1 to 4
29: dsp "Enter "&A$[L]&" Channel to Plot"
30: ent "",A[L];if not flg13;jmp 3
31: if L<3;jmp -2
32: L-1)B;jmp 5
33: if A[L]<0 or A[L]>39;dsp "No Such Channel Number";wait 1500;jmp -2
34: for M=1 to J;if C[M]#A[L];next M;jmp 2
35: M)A[L];next L;4)B;jmp 2
36: fxd 0;dsp "No Data Collected at Channel"&str(A[L]);wait 1500;jmp -7
37: for L=1 to B-1;for M=L+1 to B;if A[M]>=A[L];jmp 2
38: A[L])A;A[M])A[L];A)A[M]
39: next M;next L
40:
41: "CALCULATE UNITS FOR AXES";
42: rread 1,K+1)K,T$,C;if T$<B$;jmp 0
43: K)Y;1)F;0)H;for L=1 to B;for M=1 to A[L]-H;sread 1,C;next M
44: "TIME")H[1];C)V[L,1])V[L,2];A[L])H;next L
45: for L=K+1 to M;rread 1,L,T$,C;if T$>E$;jmp 5
46: F+1)F;0)H;for M=1 to B;for O=1 to A[M]-H;sread 1,C;next O
47: if C<V[M,1];C)V[M,1]
48: if C>V[M,2];C)V[M,2]
49: A[M])H;next M;next L
50: rread 1,L-1)Z,T$;"TIME")H[2]
51: if H[2]-H[1]>86399;"DAYS")U$;86400)D;jmp 4
52: if H[2]-H[1]>3599;"HOURS")U$;3600)D;jmp 3
53: if H[2]-H[1]>59;"MINUTES")U$;60)D;jmp 2
54: "SECONDS")U$;1)D
55: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2];H[2]-H[1])M
56: for L=1 to J;for M=1 to int(5-len(C$[L])/2)
57: " "&C$[L])C$[L];next M;next L
58: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])T$;4)K
59:
60: "INITIALIZE PLOT AREA";
61: hdcpy 1;pse 705;fxd 0;scl -7.125,0,0,10.5;plt 0,0,1
62: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
63: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
64: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
65: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.6,1;lbl "Analog Stack Plot"
66: plt -9/16,3.4,1;lbl "Data File";plt -5.5/16,3.6,1;lbl "Name"
67: plt -1/16,3.52,1;lbl F$
68: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
69: plt -1/16,5.1,1;lbl str(F)
70: plt -7.5/16,6.4,1;lbl "Frequency";plt -1/16,6.35,1
71: fxd 2;lbl str(1/P#1e-6)&" MHz "
72: plt -9/16,7.9,1;lbl "Measurement";plt -5.5/16,8,1;lbl "Interval"
73: plt -1/16,7.9,1;fxd 0;lbl str(I)&" secs"
74: plt -9/16,9.3,1;lbl "  Channel";plt -5.5/16,9.4,1;lbl "Function"
75: "DC Voltage"M$;if J$[A[1],A[1]]="A","A")M$[1,1]
76: if J$[A[1],A[1]]="O","    Ohms")M$[1,1]
77: plt -1/16,9.3,1;lbl M$;plt -1.375,10.4,1
78: if B#2;jmp 4
79: plt -4.125,10.4,2;plt -4.125,1;plt -1.375,1;plt -1.375,10.4
80: plt -4.375,10.4,1;plt -7.125,10.4,2;plt -7.125,1;plt -4.375,1
81: plt -4.375,10.4
82: if B#3;jmp 5
83: plt -3.125,10.4,2;plt -3.125,1;plt -1.375,1;plt -1.375,10.4
84: plt -3.375,10.4,1;plt -5.125,10.4,2;plt -5.125,1;plt -3.375,1
85: plt -3.375,10.4;plt -5.375,10.4,1;plt -7.125,10.4,2;plt -7.125,1
86: plt -5.375,1;plt -5.375,10.4
```

```
87: if B#4;jmp 6
88: plt -2.625,10.4,2;plt -2.625,1;plt -1.375,1;plt -1.375,10.4
89: plt -2.875,10.4,1;plt -4.125,10.4,2;plt -4.125,1;plt -2.875,1
90: plt -2.875,10.4;plt -4.375,10.4,1;plt -5.625,10.4,2;plt -5.625,1
91: plt -4.375,1;plt -4.375,10.4;plt -5.875,10.4,1;plt -7.125,10.4,2
92: plt -7.125,1;plt -5.875,1;plt -5.875,10.4
93:
94: "LABEL AXES AND PLOT DATA"
95: for D=1 to B;H[2]-10.4W/9.4)$
96: V[0,1]-(V[0,2]-V[0,1])(1.375+(D-1)('SCALE'(B)+.25))/'SCALE'(B)}R
97: scl R+7.125(V[0,2]-V[0,1])/'SCALE'(B),R,S,(10.5H[2]-.1S)/10.4
98: for L-H[1]+D to H[2]-D by D
99: plt V[0,1]+((V[0,2]-V[0,1])/'TIC'(B))C)/4,L,1;plt V[0,1],L,2;next L
100: for L-V[0,1]+C to V[0,2]+1e-6-C by C
101: plt L,H[2],1;plt L,H[2]-.104W/9.4,2;next L
102: for L-H[2]-D to H[1]+D by -D
103: plt V[0,2],L,1;plt V[0,2]-C/4,L,2;next L
104: for L-V[0,2] to V[0,1]-1e-6 by -C
105: plt L-C/8,H[1]-.42W/9.4,1;csiz 1,2,1,90;fxd 3;lbl str(L)
106: if L#V[0,1] and L#V[0,2];plt L,H[1],1;plt L,H[1]+.104W/9.4,2
107: next L;csiz 1.5,2,1,180
108: plt V[0,1]+(1-.75/'SCALE'(B))(V[0,2]-V[0,1])/2,H[1]-.55''/9.4,1
109: lbl C$[A[0]];rread 1,Y,T$,S
110: for M=1 to A[0];rread 1,S;next M
111: plt S,'TIME',1;cplt -.33,-.15;lbl "o";plt S,'TIME',1
112: for L=Y+1 to Z;rread 1,L,T$,S
113: for M=1 to A[0];rread 1,S;next M
114: plt S,'TIME',2;next L
115: cplt -.33,-.15;lbl "o"
116: next D;csiz 1,2,1,90
117: scl 7.125,0,H[2]-10.4W/9.4,(10.5H[2]-.1(H[2]-10.4W/9.4))/10.4
118: for L=H[1] to H[2] by D;plt 1.125,L-.104W/9.4,1;L)@
119: for M=1 to 12
120: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
121: if Q-2764800<0;gto "PRINT"
122: Q-2678400)@
123: if not (M=4 or M=6 or M=9 or M=11);jmp 3
124: if Q-2678400<0;gto "PRINT"
125: Q-2592000)@
126: if not M=2;jmp 3
127: if Q-2505600<0;gto "PRINT"
128: Q-2419200)@
129: next M
130: "PRINT";if D=86400;fxd 0;lbl str(Q/D)
131: if D=3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
132: if D=60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
133: if D=1;fxd 0;lbl str((Q-60int(Q/60))/D)
134: next L;csiz 1.5,2,1,90;plt .875,H[1]+.329787W,1
135: if B$="0000000000";jmp 2
136: B$[1,2]&":"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M$;jmp 2
137: rread 1,5,T$;T$[1,2]&":"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
138: lbl "Time in "&U$&" beginning  "&M$;wtb"PRINTER",12,13,27,69
139: end
140:
141: "TIME"
142: 0)Q;for M=1 to val(T$[1,2])-1
143: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;jmp 3
144: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;jmp 2
145: if M=2;Q+2419200)@
146: next M
147: Q+86400val(T$[3,4])+3600val(T$[5,6])}@
148: ret Q+60val(T$[7,8])+val(T$[9,10])
149:
150: "SCALE"
151: if pi=2;ret 2.75
152: if pi=3;ret 1.75
153: if pi=4;ret 1.25
154:
155: "TIC"
156: if pi=2;ret 10
157: if pi=3;ret 6
158: if pi=4;ret 4
*8544
```

```
0: "PROGRAM:  ANALOG DATA STACK DISPLAY"
1:
2: "*************************************************************"
3: "*  Kevin Miller ,Trip Carter  *  File: STK18                 *"
4: "*  July 21, 1983              *  Update: July 21, 1983      *"
5: "*  Naval Research Laboratory  *  Output Device: 7245        *"
6: "*  Space Applications Branch  *  Select Code: 718          *"
7: "*************************************************************"
8:
9: "GET FILE NAME AND PLOT INTERVAL"
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PRINTER",706,"PLOTTER",705,"CLOCK",9
12: dim F$[6],D$[6],T$[14],U$[7],M$[40],H[2],V[4,2],B$[12],E$[12]
13: dim A[4],A$[4,3],C[26],C$[25,10],J$[25]
14: dim C,D,E,F,G,I,J,K,L,M,N,P,Q,X,Y,Z,4)K
15: "1st")A$[1];"2nd")A$[2];"3rd")A$[3];"4th")A$[4]
16: "")F$;ent "Enter Data File Name",F$;if flg13;jmp 8
17: asgn F$,1,1,8
18: if 0=1;dsp F$&" does not exist";wait 1500;jmp -2
19: asgn F$,1,1;rread 1,1,G,N,I,D$,P,J
20: if N<3;dsp "Not Enough Data for Plot";stp
21: for L=1 to J;rread 1,C[L],C$[L],J$[L,L];next L
22: ent "Plot Data from:  MMDDHHMMSS",B$
23: if flg13;"0000000000")B$
24: if len(B$)<10;dsp "Not in Proper Form";wait 1500;jmp -2
25: ent "Plot Data to:  MMDDHHMMSS",E$
26: if flg13;"9999999999")E$
27: if len(E$)≠10;dsp "Not in Proper Form";wait 1500;jmp -2
28: for L=1 to 4
29: dsp "Enter "&A$[L]&" Channel to Plot"
30: ent "",A[L];if not flg13;jmp 3
31: if L<3;jmp -2
32: L-1)B;jmp 5
33: if A[L]<8 or A[L]>39;dsp "No Such Channel Number";wait 1500;jmp -2
34: for M=1 to J;if C[M]≠A[L];next M;jmp 2
35: M)A[L];next L;4)B;jmp 2
36: fxd 0;dsp "No Data Collected at Channel"&str(A[L]);wait 1500;jmp -7
37: for L=1 to B-1;for M=L+1 to B;if A[M]>=A[L];jmp 2
38: A[L])A;A[M])A[L];A)A[M]
39: next M;next L
40:
41: "CALCULATE UNITS FOR AXES"
42: rread 1,K+1)K,T$,C;if T$<B$;jmp 0
43: K)Y;1)F;0)H;for L=1 to B;for M=1 to A[L]-H;sread 1,C;next M
44: "TIME")H[1];C)V[L,1])V[L,2];A[L])H;next L
45: for L=K+1 to N;rread 1,L,T$,C;if T$>E$;jmp 5
46: F+1)F;0)H;for M=1 to B;for D=1 to A[M]-H;sread 1,C;next 0
47: if C<V[M,1];C)V[M,1]
48: if C>V[M,2];C)V[M,2]
49: A[M])H;next M;next L
50: rread 1,L-1)Z,T$;"TIME")H[2]
51: if H[2]-H[1]>86399;"DAYS")U$;86400)D;jmp 4
52: if H[2]-H[1]>3599;"HOURS")U$;3600)D;jmp 3
53: if H[2]-H[1]>59;"MINUTES")U$;60)D;jmp 2
54: "SECONDS")U$;1)D
55: Dint(H[1]/D))H[1];Dint(H[2]/D+1))H[2];H[2]-H[1])M
56: for L=1 to J;for M=1 to int(5-len(C$[L])/2)
57: " "&C$[L])C$[L];next M;next L
58: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])T$;4)K
59:
60: "INITIALIZE PLOT AREA"
61: fxd 0;hdcpy 0;ipse 718;pclr;scl 0,10.5,0,7.125;plt 0,0,1
62: plt 0,.75,2;plt 10.4,.75;plt 10.4,0;plt 0,0;plt 0,5/16,1;plt 10.4,5/16,2
63: for L=1.5 to 9 by 1.5;plt L,.75,1;plt L,0,2;next L;csiz 1.5,2,1
64: plt .55,9/16,1;lbl "Date";plt .5,5.7/16,1;lbl "Today"
65: plt .21,1/11,1;lbl T$[1,2]&"-"&T$[4,5]&"-80 "
66: plt 1.85,7.5/16,1;lbl "Program";plt 1.61,1/11,1;lbl " Stackplot"
67: plt 3.25,9/16,1;lbl "Data File";plt 3.57,5.7/16,1;lbl "Name"
68: plt 3.52,1/11,1;lbl F$&" "
69: plt 4.88,9/16,1;lbl "Points";plt 4.83,5.7/16,1;lbl "Plotted"
70: plt 4.92,1/11,1;lbl str(F)&" "
71: plt 6.2,7.5/16,1;lbl "Frequency";plt 6.15,1/11,1
72: fxd 2;lbl str(1/P*1e-6)&" MHz "
73: plt 7.6,9/16,1;lbl "Measurement";plt 7.8,5.7/16,1;lbl "Interval"
74: plt 7.6,1/11,1;fxd 0;lbl str(I)&" secs "
75: plt 9.18,9/16,1;lbl "Channel";plt 9.18,5.7/16,1;lbl "Function"
76: "DC Voltage")M$;if J$[A[1],A[1]]="A";"A")M$[1,1]
77: if J$[A[1],A[1]]="Ω";"  Ohms")M$
78: plt 9.1,1/16,1;lbl M$;plt 10.4,1.375,1
79: if B≠2;jmp 4
80: plt 10.4,4.125,2;plt 1,4.125;plt 1,1.375;plt 10.4,1.375
81: plt 10.4,4.375,1;plt 10.4,7.125,2;plt 1,7.125;plt 1,4.375
82: plt 10.4,4.375
83: if B≠3;jmp 5
84: plt 10.4,3.125,2;plt 1,3.125;plt 1,1.375;plt 10.4,1.375
85: plt 10.4,3.375,1;plt 10.4,5.125,2;plt 1,5.125;plt 1,3.375
86: plt 10.4,3.375;plt 10.4,5.375,1;plt 10.4,7.125,2;plt 1,7.125
```

```
87: plt 1,5.375;plt 10.4,5.375
88: if B#4;jmp 6
89: plt 10.4,2.625,2;plt 1,2.625;plt 1,1.375;plt 10.4,1.375
90: plt 10.4,2.875,1;plt 10.4,4.125,2;plt 1,4.125;plt 1,2.875
91: plt 10.4,2.875;plt 10.4,4.375,1;plt 10.4,5.625,2;plt 1,5.625
92: plt 1,4.375;plt 10.4,4.375;plt 10.4,5.875,1;plt 10.4,7.125,2
93: plt 1,7.125;plt 1,5.875;plt 10.4,5.875
94:
95: 'LABEL AXES AND PLOT DATA';
96: for D=1 to 8;H[2]-10.4W/9.4)$
97: V[0,1]-(V[0,2]-V[0,1])(1.375+(D-1)('SCALE'(B)+.25))/'SCALE'(B))R
98: sel S,(10.5H[2]-.15)/10.4,R,R+7.125(V[0,2]-V[0,1])/'SCALE'(B)
99: for L-H[1]+D to H[2]-D by D
100: plt L,V[0,1]+((V[0,2]-V[0,1])/'TIC'(B))C)/4,1;plt L,V[0,1],2;next L
101: for L-V[0,1]+C to V[0,2]+1e-6-C by C
102: plt H[2],L,1;plt H[2]-.104W/9,L,2;next L
103: for L-H[2]-D to H[1]+D by -D
104: plt L,V[0,2],1;plt L,V[0,2]-C/4,2;next L
105: for L-V[0,2] to V[0,1]-1e-6 by -C
106: plt H[1]-.68W/9.4,L-C/8,1;csiz 1,2,1,0;fxd 2;lbl str(L)
107: if L#V[0,1] and L#V[0,2];plt H[1],L,1;plt H[1]+.104W/9.4,L,2
108: next L;csiz 1.5,2,1,90
109: plt H[1]-.85W/9.4,V[0,1]+(1-.75/'SCALE'(B))(V[0,2]-V[0,1])/9,1
110: lbl C$[A[0]];rread 1,Y,T$,S
111: for M=1 to A[0];sread 1,S;next M
112: plt 'TIME',S,1;cplt -.15,-.33;lbl "e";plt 'TIME',S,1
113: for L-Y+1 to Z;rread 1,L,T$,S
114: for M=1 to A[0];sread 1,S;next M
115: plt 'TIME',S,2;next L
116: cplt -.15,-.33;lbl "e"
117: next D;csiz 1,2,1,0
118: sel H[2]-10.4W/9.4,(10.5H[2]-.1(H[2]-10.4W/9.4))/10.4,0,7.125
119: for L-H[1] to H[2] by D;plt L-.177W/9.4,1.125,1;L)@
120: for M=1 to 12
121: if not (M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12);jmp 3
122: if Q-2764800<0;gto 'PRINT'
123: Q-2678400)@
124: if not (M=4 or M=6 or M=9 or M=11);jmp 3
125: if Q-2678400<0;gto 'PRINT'
126: Q-2592000)@
127: if not M=2;jmp 3
128: if Q-2505600<0;gto 'PRINT'
129: Q-2419200)@
130: next M
131: 'PRINT';if D=86400;fxd 0;lbl str(Q/D)
132: if D=3600;fxd 0;lbl str((Q-86400int(Q/86400))/D)
133: if D=60;fxd 0;lbl str((Q-3600int(Q/3600))/D)
134: if D=1;fxd 0;lbl str((Q-60int(Q/60))/D)
135: next L;csiz 1.5,2,1,0;plt H[1]+.2378W,.875,1
136: if B$='0000000000';jmp 2
137: B$[1,2]&"-"&B$[3,4]&"  "&B$[5,6]&":"&B$[7,8]&":"&B$[9,10])M$;jmp 2
138: rread 1,5,T$;T$[1,2]&"-"&T$[3,4]&"  "&T$[5,6]&":"&T$[7,8]&":"&T$[9,10])M$
139: lbl 'Time is "&U$&" beginning  "&M$&" "
140: end
141:
142: 'TIME';
143: 0)Q;for M=1 to val(T$[1,2])-1
144: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;jmp 3
145: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;jmp 2
146: if M=2;Q+2419200)Q
147: next M
148: Q+86400val(T$[3,4])+3600val(T$[5,6]))Q
149: ret Q+60val(T$[7,8])+val(T$[9,10])
150:
151: 'SCALE';
152: if p1=2;ret 2.75
153: if p1=3;ret 1.75
154: if p1=4;ret 1.25
155:
156: 'TIC';
157: if p1=2;ret 10
158: if p1=3;ret 6
159: if p1=4;ret 4
#25
```

9.32 Program: QUI16

```
0: "PROGRAM:  QUICK LIST";
1:
2: "*************************************************************";
3: "*  Kevin Miller                 *  File: QUI16                 *";
4: "*  January 14, 1980             *  Update: December 20, 1980   *";
5: "*  Naval Research Laboratory    *  Output Device: 9825         *";
6: "*  Space Applications Branch    *  Select Code: 16             *";
7: "*************************************************************";
8:
9: dim F$[6],D$[6],M$[40],G,M,I,J
10: ""]F$;ent "Enter Data File Name",F$;if flg13;jmp 0
11: assn F$,1,1,0
12: if Q=1;dsp F$&" does not exist";wait 1500;jmp -2
13: assn F$,1,1;rread 1,1,G,M,I,D$,P,J;spc 3
14: fxd 2;"----------------"]M$;prt M$,"Gate Time in","msec:",G,M$
15: fxd 0;prt "Total Number of","Points","Collected:",M
16: fxd 0;prt M$,"Time Interval","Between","Measurements"
17: prt "in secs:",I,M$,"Date:    "&D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6],M$
18: flt 6;prt "Signal Period:",P,M$
19: fxd 0;prt "Voltage Channels","Sampled:",J,M$;spc 3
20: end
=12728
```

9.33 Program: .QUI16

```
0: "SUBPROGRAM:  QUICK LIST";
1:
2: "*************************************************************";
3: "*  Kevin Miller                 *  File: .QUI16                *";
4: "*  January 14, 1980             *  Update: December 20, 1980   *";
5: "*  Naval Research Laboratory    *  Output Device: 9825         *";
6: "*  Space Applications Branch    *  Select Code: 16             *";
7: "*************************************************************";
8:
9: fxd 2;"----------------"]M$;prt M$,"Gate Time in","msec:",G,M$
10: fxd 0;prt "Total Number of","Points to be","Collected:",M
11: fxd 0;prt M$,"Time Interval","Between","Measurements"
12: prt "in secs:",I,M$,"Date:    "&D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6],M$
13: flt 6;prt "Signal Period:",P,M$,"Points Collected"
14: fxd 0;prt "So Far:",K-1,M$,"Voltage Channels","Sampled:",J,M$;spc 3
15: ret
=18685
```

3.34 Program: SCA06

```
0: "PROGRAM:  VOLTAGE DATA LIST"
1:
2: "*************************************************"
3: "*  Kevin Miller                 *  File: SCA06     *"
4: "*  January 14, 1988             *  Update: December 28, 1988  *"
5: "*  Naval Research Laboratory    *  Output Device: 7245        *"
6: "*  Space Applications Branch    *  Select Code: 706           *"
7: "*************************************************"
8:
9: "GET FILE NAME AND LIST INTERVAL"
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "PRINTER",706)r0,"CLOCK",9
12: dim F$[6],D$[6],T$[12],R$[12],C[25],C$[40,10],J$[25],S[15],M[15,2],M$[40]
13: "")F$;ent "Enter Data File Name",F$;if flg13;jmp 0
14: asgn F0,1,1,0;if 0#1;jmp 2
15: dsp F$&" does not exist";wait 1500;gto -2
16: asgn F0,1,1;rread 1,1,G,N,I,D$,P,J
17: if J=0;spe 3;prt "No VOLTAGE DATA","Obtained this","Test.";spe 3;end
18: for L=1 to J;rread 1,C[L],C$[L],J$[L,L];next L
19: ent "List Data Since:  MMDDHHMMSS",T$
20: if flg13;"0000000000")T$
21: if len(T$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
22:
23: "FORMAT STATEMENTS"
24: fmt 2,2/,21x,"   ***** Thermal Vacuum Test Facility *****   ",2/
25: fmt 3,c14,z
26: fmt 4,c10,z
27: fmt 5,"**************",z
28: fmt 6,"**********",z
29: fmt 7,f9.3,z
30: fmt 8,c5,x,z
31: fmt 9,c8,z
32:
33: "ADDITIONAL COLUMNS"
34: 0)A;ent "How Many Additional Columns ?",A
35: if A>15;dsp "Too Many";wait 1500;gto -1
36: if A<=0;gto "PRINT BOX"
37: ent "Print Present Channel Names ?",R$
38: if cap(R$[1,1])#"Y";gto +5
39: fxd 0;prt " ","  ","  Channel Names"," ";for X=1 to J;if C[X]<10;gto +2
40: prt str(C[X])&") "&C$[X];gto +2
41: prt " "&str(C[X])&") "&C$[X]
42: next X;prt " "," "," "
43: for L=1 to A;fxd 0;dsp "Name of Add-on Column"&str(L);ent "",C$[J+L]
44: if flg13;"Add-on "&str(L))C$[J+L]
45: 50)M[L,1];ent "Origin Channel ?",M[L,1]
46: for X=1 to J;if M[L,1]=C[X];X)M[L,1];gto +3
47: next X;if M[L,1]=50;gto +2
48: dsp "Invalid Column Number";wait 1500;gto -3
49: 50)M[L,2];ent "Multiplicative Channel ?",M[L,2]
50: for X=1 to J;if M[L,2]=C[X];X)M[L,2];gto +3
51: next X;if M[L,2]=50;gto +2
52: dsp "Invalid Column Number";wait 1500;gto -3
53: 1)S[L];ent "Scale Factor ?",S[L];next L
54:
55: "PRINT BOX"
56: wtb "PRINTER",12,13
57: cll "shtab"(17,29,41,53,65,77)
58: wrt "PRINTER.2"
59: fmt 1,40"*",/,"*",38" ","*";wrt "PRINTER.1"
60: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])M$;gsb "PAD"
61: wrt "PRINTER","*  Start Date of Test:     "&M$&"  *"
62: fxd 2;str(1/P#1e-6)&" MHz")M$;gsb "PAD"
63: wrt "PRINTER","*  Signal Frequency:       "&M$&"  *"
64: fxd 0;str(G)&" msec")M$;gsb "PAD"
65: wrt "PRINTER","*  Gate Time:              "&M$&"  *"
66: fxd 0;str(N))M$;gsb "PAD"
67: wrt "PRINTER","*  Number of Points:       "&M$&"  *"
68: fxd 0;str(I)&" secs")M$;gsb "PAD"
69: wrt "PRINTER","*  Measurement Interval:   "&M$&"  *"
70: F$)M$;gsb "PAD"
71: wrt "PRINTER","*  Data File Name:          "&M$&" *"
72: "SCA06")M$;gsb "PAD"
73: wrt "PRINTER","*  Program:                 "&M$&" *"
74: fmt 1,"*",38" ","*",/,40"*";wrt "PRINTER.1";jmp 3
75: "PAD";for L=1 to 12;if M$[L,L]="";" ")M$[L,L]
76: next L;ret
77: wtb "PRINTER",10,10
78: N)S;0)E;1)r51
79:
80: "CENTER CHANNEL NAMES IN C$"
81: for L=1 to J+A
82: len(C$[L])))D;int((11-D)/2))R
83: for M=D+1 to 10
84: " ")C$[L,M,M]
85: next M
86: for M=D+R to R+1 by -1
```

```
87: C$[L,M-P,M-R])C$[L,M,M]
88: next M
89: for M=1 to R
90: " ")C$[L,M,M]
91: next M
92: next L
93:
94: "PRINT COLUMN HEADERS"
95: 0)R,J+A-E)D;if D>S;D-S)R;S)D
96: wrt "PRINTER.3","  Date    Time  "
97: cll "tab"
98: wrt "PRINTER.4","Phase    "
99: for L=E+1 to E+D
100: cll "tab"
101: wrt "PRINTER.4",C$[L]
102: next L
103: wtb "PRINTER",13,10
104: wrt "PRINTER.5"
105: for L=1 to D+1
106: cll "tab"
107: wrt "PRINTER.6"
108: next L
109: wtb "PRINTER",10,13
110:
111: "PRINT DATA"
112: for L=1 to S;rread 1,L+4,R$,r1
113: if R$<T$;gto +13
114: for M=1 to J;rread 1,r(M+1);next M
115: for X=1 to A;S[X]#r(M[X,1]+1)#r(M[X,2]+1))r(J+X+1);next X
116: wrt "PRINTER.8",R$[1,2]&"-"&R$[3,4]
117: wrt "PRINTER.9",R$[5,6]&":"&R$[7,8]&":"&R$[9,10]
118: cll "tab"
119: fmt 2,e10.4,z
120: wrt "PRINTER.2",r1
121: for M=E+1 to E+D
122: cll "tab"
123: wrt "PRINTER.7",r(M+1)
124: next M
125: wtb "PRINTER",10,13
126: next L
127: wtb "PRINTER",12,13
128: if R#0;E-5)E;gto "PRINT COLUMN HEADERS"
129: wtb "PRINTER",27,69
130: end
131:
132: "tab"
133: if p2=0;gto +4
134: if p2<0;gto +2
135: wtb r0,11;jmp 2((p2-1)p2)-0)
136: wtb r0,27,56;jmp (p2+1)p2)-0
137: if p1=0;1)p1
138: if p1<0;gto +2
139: wtb r0,9;jmp 2((p1-1)p1)-0)
140: wtb r0,27,52;jmp (p1+1)p1)-0
141: ret
142: "shtab"
143: 1)Y;wtb r0,13;if (p1)Q)-0;ret
144: if (Q-1)Q)>0;wtb r0,32;jmp 0
145: wtb r0,27,48
146: if p(Y+1)Y)-0;wtb r0,13;ret
147: pY-p(Y-1)+1)0;gto -3
#7602
```

3.35 Program: SCA18

```
0: "PROGRAM:  VOLTAGE DATA DISPLAY";
1:
2: "*************************************************";
3: "*  Kevin Miller, Trip Carter  *  File: SCA18     *";
4: "*  July 25,1983               *  Update: July 25,1983 *";
5: "*  Naval Research Laboratory  *  Output Device: 7245 *";
6: "*  Space Applications Branch  *  Select Code: 718   *";
7: "*************************************************";
8:
9: "GET FILE NAME AND LIST INTERVAL";
10: dsp "Insert Softcopy Graphics Tape";stp
11: ldb 1;dev "SCREEN",718)r0,"CLOCK",9
12: dim F$[6],D$[6],T$[12],R$[12],C[25],C$[40,10],J$[25],S[15],M[15,2],M$[40]
13: ""]F$;ent "Enter Data File Name",F$;if flg13;jmp 8
14: asgn F$,1,1,0;if u#1;jmp 2
15: dsp F$&" does not exist";wait 1500;gto -2
16: asgn F$,1,1;rread 1,1,G,M,I,D$,F,J
17: if J=0;spc 3;prt "No VOLTAGE DATA","Obtained this","Test.";spc 3;end
18: for L=1 to J;rread 1,C[L],C$[L],J$[L,L];next L
19: ent "List Data Since:  MMDDHHMMSS",T$
20: if flg13;"0000000000"]T$
21: if len(T$)#10;dsp "Not in Proper Form";wait 1500;jmp -2
22:
23: "ADDITIONAL COLUMNS";
24: 0)A;ent "How Many Additional Columns ?",A
25: if A>15;dsp "Too Many";wait 1500;gto -1
26: if A<=0;gto "PRINT BOX"
27: ent "Print Present Channel Names ?",R$
28: if cap(R$[1,1])#"Y";gto +5
29: fxd 0;prt " ","  "," Channel Names"," ";for X=1 to J;if C[X]<10;gto +2
30: prt str(C[X])&") "&C$[X];gto +2
31: prt " "&str(C[X])&") "&C$[X]
32: next X;prt " "," "," "
33: for L=1 to A;fxd 0;dsp "Name of Add-on Column"&str(L);ent "",C$[J+L]
34: if flg13;"Add-on "&str(L))C$[J+L]
35: 50)M[L,1];ent "Origin Channel ?",M[L,1]
36: for X=1 to J;if M[L,1]=C[X];X)M[L,1];gto +3
37: next X;if M[L,1]=50;gto +2
38: dsp "Invalid Column Number";wait 1500;gto -3
39: 50)M[L,2];ent "Multiplicative Channel ?",M[L,2]
40: for X=1 to J;if M[L,2]=C[X];X)M[L,2];gto +3
41: next X;if M[L,2]=50;gto +2
42: dsp "Invalid Column Number";wait 1500;gto -3
43: 1)S[L];ent "Scale Factor ?",S[L];next L
44:
45: "PRINT BOX";
46: fxd 3;hdcpy 0;psc 718;pclr
47: scl 1,83,-49,1;plt 19,-3,1;wtb "SCREEN","cs0,,,pe1"
48: wtb "SCREEN","cs0,,,pe1,,,tx ##### THERMAL VACUUM",3,13,10
49: wtb "SCREEN","cs0,,,pe1,,,tx TEST FACILITY #####",3,13,10
50: wrt "SCREEN","pe0"
51: plt 1,-8;wtb "SCREEN","cs0,,,pe1,,,tx ##########################",3,13,10
52: wtb "SCREEN","cs0,,,pe1,,,tx##########",3,13,10
53: wrt "SCREEN","pe0";plt 1,-9
54: wtb "SCREEN","cs0,,,pe1,,,tx #          ",3,13,10
55: wtb "SCREEN","cs0,,,pe1,,,tx                                    #",3,13,10
56: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])M$;gsb "PAD"
57: wrt "SCREEN","pe0";plt 1,-10
58: wtb "SCREEN","cs0,,,pe1,,,tx #  Start Date of Test:      "&M$&"#",3,13,10
59: fxd 2;str(1/P#1e-6)&" MHz")M$;gsb "PAD"
60: wrt "SCREEN","pe0";plt 1,-11
61: wtb "SCREEN","cs0,,,pe1,,,tx #  Signal Frequency:       "&M$&" #",3,13,10
62: fxd 0;str(G)&" msec")M$;gsb "PAD"
63: wrt "SCREEN","pe0";plt 1,-12
64: wtb "SCREEN","cs0,,,pe1,,,tx #  Gate Time:             "&M$&" #",3,13,10
65: fxd 0;str(N)]M$;gsb "PAD"
66: wrt "SCREEN","pe0";plt 1,-13
67: wtb "SCREEN","cs0,,,pe1,,,tx #  Number of Points:      "&M$&" #",3,13,10
68: fxd 0;str(I)&" secs")M$;gsb "PAD"
69: wrt "SCREEN","pe0";plt 1,-14
70: wtb "SCREEN","cs0,,,pe1,,,tx #  Measurement Interval:  "&M$&" #",3,13,10
71: F$)M$;gsb "PAD"
72: wrt "SCREEN","pe0";plt 1,-15
73: wtb "SCREEN","cs0,,,pe1,,,tx #  Data File Name:         "&M$&"#",3,13,10
74: "SCA06")M$;gsb "PAD"
75: wrt "SCREEN","pe0";plt 1,-16
76: wtb "SCREEN","cs0,,,pe1,,,tx #  Program:               "&M$&"#",3,13,10
77: wrt "SCREEN","pe0";plt 1,-17
78: wtb "SCREEN","cs0,,,pe1,,,tx #          ",3,13,10
79: wtb "SCREEN","cs0,,,pe1,,,tx                                    #",3,13,10
80: wrt "SCREEN","pe0";plt 1,-18
81: wtb "SCREEN","cs0,,,pe1,,,tx ################",3,13,10
82: wtb "SCREEN","cs0,,,pe1,,,tx##########################",3,13,10;jmp 3
83: "PAD";for L=1 to 12;if M$[L,L]="";" ")M$[L,L]
84: next L;ret
85: N)S;0)E;1)r51;22)K
86: 0)R;J+A-E)D;if D>5;D-5)R;5)B
```

```
87: gsb "PRINT COLUMN HEADERS"
88:
89: "PRINT DATA":
90: for L=1 to S;rread 1,L*4,R$,r1
91: if R$<T$;gto *17
92: for M=1 to J;rread 1,r(M*1);next M
93: for X=1 to A;S(X)=r(M[X,1)*1)=r(M[X,2)*1))r(J*X*1);next X
94: wrt "SCREEN","pe0";plt 1,-K
95: wtb "SCREEN","cs0,;;pe1,;;tz"&R$(1,2)&"-"&R$(3,4),3,13,10
96: wtb "SCREEN","cs0,;;pe1,;;tz "&R$(5,6)&":"&R$(7,8)&":"&R$(9,10),3,13,10
97: wrt "SCREEN","pe0,"
98: fit 4;str(r1))M$;for X=len(M$)+1 to 9;" "&M$)M$;next X
99: plt 15,-K;wtb "SCREEN","cs0,;;pe1,;;tz"&M$,3,13,10
100: wrt "SCREEN","pe0,"
101: for M=E+1 to E+D
102: fxd 3;str(r(M*1)))M$;for X=len(M$)+1 to 9;" "&M$)M$;next X
103: plt 27+11(M-E-1),-K;wtb "SCREEN","cs0,;;pe1,;;tz"&M$,3,13,10
104: wrt "SCREEN","pe0,";next M
105: K+2)K;if K<50 and L<S;gto *3
106: dsp "Press Continue";stp
107: pclr;0)K;gsb "PRINT COLUMN HEADERS"
108: next L
109: if R=0;end
110: E+5)E;0)R;J+A-E)D;if D>5;D-5)R;5)D
111: gto "PRINT DATA"
112:
113: "CENTER CHANNEL NAMES IN C$":
114: for L=1 to J+A
115: len(C$[L]))D;int((11-D)/2))R
116: for M=D+1 to 10
117: " ")C$[L,M,M]
118: next M
119: for M=D+R to R+1 by -1
120: C$[L,M-R,M-R])C$[L,M,M]
121: next M
122: for M=1 to R
123: " ")C$[L,M,M]
124: next M
125: next L;ret
126:
127: "PRINT COLUMN HEADERS":
128: wrt "SCREEN","pe0,";plt 1,-K
129: wtb "SCREEN","cs0,;;pe1,;;tz Date    Time",3,13,10
130: wrt "SCREEN","pe0,";plt 19,-K;wtb "SCREEN","cs0,;;pe1,;;tzPhase",3,13,10
131: for B=E+1 to E+D;wrt "SCREEN","pe0,"
132: plt 27+11(B-E-1),-K;wtb "SCREEN","cs0,;;pe1,;;tz "&C$[B],3,13,10
133: next B;K+1)K
134: wrt "SCREEN","pe0";plt 1,-K
135: wtb "SCREEN","cs0,;;pe1,;;tz##############",3,13,10
136: for B=1 to D+1;wrt "SCREEN","pe0,"
137: plt 16+11(B-1),-K;wtb "SCREEN","cs0,;;pe1,;;tz##########",3,13,10
138: next B
139: K+2)K;ret
#11122
```

9.36 Program: .SCA06

```
0: "SUBPROGRAM:  LIST VOLTAGE CHANNEL DATA"&
1:
2: "*************************************************************"&
3: "*  Kevin Miller                  *  File: .SCA06               *"&
4: "*  January 14, 1989              *  Update: December 28, 1988  *"&
5: "*  Naval Research Laboratory     *  Output Device: 7245        *"&
6: "*  Space Applications Branch     *  Select Code: 706           *"&
7: "*************************************************************"&
8:
9: "FULL OR PARTIAL LIST"&
10: if J#0;jmp 3
11: spc 3;prt "No VOLTAGE DATA","will be Obtained"
12: prt "this Test.";spc 3;gto "RETURN"
13: spc 3;prt "---------------","  VOLTAGE LIST","---------------";spc 2
14: prt "Kevin Special","  Function Key:";spc
15: prt "f0 to list ALL","    data","f1 to list data"
16: prt "    collected","    since 12:00","    Midnight";spc 3
17: "CYCLE";if flg0;0)V;cfg 0;jmp 3
18: if flg1;1)V;wrt "CLOCK","R";red "CLOCK",M$;M$[1,2]&M$[4,5])U$;cfg 1;jmp 2
19: gto "CYCLE"
20:
21: "PRINT BOX"&
22: fmt 3,c14,z;fmt 4,c10,z;fmt 5,"##############",z
23: fmt 6,"##########",z;fmt 7,f9.3,z;fmt 8,c5,x,z;fmt 9,c8,z
24: wtb "PRINTER",13;for M=1 to 16;wtb "PRINTER",32;next M
25: wtb "PRINTER",27,49;for L=1 to 5;for M=1 to 12;wtb "PRINTER",32;next M
26: wtb "PRINTER",27,49;next L;wtb "PRINTER",13,10,10
27: "  Thermal Vacuum Test Facility  ")M$
28: wrt "PRINTER","                                  #####"&M$&"#####"
29: fmt 1,40"#",/,"#",38" ","#";wtb "PRINTER",10,10,13;wrt "PRINTER.1"
30: D$[1,2]&"-"&D$[3,4]&"-"&D$[5,6])M$;gsb "PAD"
31: wrt "PRINTER","#  Start Date:              "&M$&"  #"
32: fxd 2;str(1/P#1e-6)&" MHz")M$;gsb "PAD"
33: wrt "PRINTER","#  Signal Frequency:        "&M$&"  #"
34: fxd 0;str(G)&" msec")M$;gsb "PAD"
35: wrt "PRINTER","#  Gate Time:               "&M$&"  #"/
36: fxd 0;str(K-1)Z)M$;gsb "PAD"
37: wrt "PRINTER","#  Number of Points:        "&M$&"  #"
38: fxd 0;str(I)&" secs")M$;gsb "PAD"
39: wrt "PRINTER","#  Measurement Interval:    "&M$&"  #"
40: F$)M$;gsb "PAD"
41: wrt "PRINTER","#  Data File Name:          "&M$&"  #"
42: ".SCA06")M$;gsb "PAD"
43: wrt "PRINTER","#  Program:                 "&M$&"  #"
44: fmt 1,"#",38" ","#",/,40"#";wrt "PRINTER.1";wtb "PRINTER",10,10;0)E;jmp 3
45: "PAD";for L=1 to 12;if M$[L,L]="";" ")M$[L,L]
46: next L;ret
47:
48: "PRINT COLUMN HEADERS"&
49: 0)S;J-E)D;if D)S;D-S)S;S)0
50: wrt "PRINTER.3","  Date    Time  ";wtb "PRINTER",9
51: wrt "PRINTER.4","Phase    ";for L=E+1 to E+D;wtb "PRINTER",9
52: C$[L])R$;for M=1 to S-int(len(R$)/2);R$&" ")R$;next M
53: wrt "PRINTER.4",R$;next L;wtb "PRINTER",10,13;wrt "PRINTER.5"
54: for L=1 to D+1;wtb "PRINTER",9;wrt "PRINTER.6";next L
55: wtb "PRINTER",10,13
56:
57: "PRINT DATA"&
58: for L=1 to Z;rread 1,L+4,R$;if V and R$<U$&"000000";next L
59: rread 1,L+4,R$,R;for M=1 to J;sread 1,M[M];next M
60: wrt "PRINTER.8",R$[1,2]&"-"&R$[3,4]
61: wrt "PRINTER.9",R$[5,6]&":"&R$[7,8]&":"&R$[9,10]
62: wtb "PRINTER",9;fmt 1,e10.4,z;wrt "PRINTER.1",R
63: for M=E+1 to E+D;wtb "PRINTER",9;wrt "PRINTER.7",M[M];next M
64: wtb "PRINTER",10,13;next L;wtb "PRINTER",12
65: if S#0;E+5)E;gto "PRINT COLUMN HEADERS"
66: wtb "PRINTER",27,69
67: "RETURN";ret
#30232
```

9.37 Program: USR05

```
0: "PROGRAM:  GENERAL PURPOSE USER ENTRY PLOT"
1:
2: "*************************************************************";
3: "*  Kevin Miller              *  File: USR05                  *";
4: "*  July 28, 1982             *  Update: July 28, 1982        *";
5: "*  Naval Research Laboratory *  Output Device: 7245         *";
6: "*  Space Applications Branch *  Select Code: 705            *";
7: "*************************************************************";
8:
9: "GET AXIS LIMITS AND DATA";
10: dev "PRINTER",706,"PLOTTER",705,"CLOCK",9;flt 11
11: dim A,B,G[500,2],H[2],V[2]
12: dim C$[12],F$[8],H$[80],M$[90],N$[20],V$[80],S$[4,2],T$[12],Y$[3]
13: "st")S$[1];"nd")S$[2];"rd")S$[3];"th")S$[4]
14: ent "Enter X Min",H[1]
15: if H[1]<-999.9999;dsp "Value Must Be Greater Than -1000";wait 1500;;jmp -1
16: ent "Enter X Max",H[2]
17: if H[2]>999.9999;dsp "Value Must Be Less Than 1000";wait 1500;;jmp -1
18: if H[1]>-H[2];dsp "Invalid Entry";wait 1500;;jmp -4
19: ent "Number of Horizontal Tic Marks",M
20: if (int(M))M)>40;dsp "Must Be Less Than 41";wait 1500;;jmp -1
21: ent "Number of X Axis Decimal Places",B;int(B)}B
22: if B)2;dsp "Value Must Be Less Than 3";wait 1500;;jmp -1
23: ent "Enter X Axis Label (70 Char Max)",H$
24: if len(H$))70;dsp "Label Too Long";wait 1500;;jmp -1
25: H$)M$;70)Z;gsb "CENTER"
26: M$)H$;ent "Enter Y Min",V[1]
27: if V[1]<-999.9999;dsp "Value Must Be Greater Than -1000";wait 1500;;jmp -1
28: ent "Enter Y Max",V[2]
29: if V[2]>999.9999;dsp "Value Must Be Less Than 1000";wait 1500;;jmp -1
30: if V[1]>-V[2];dsp "Invalid Entry";wait 1500;;jmp -4
31: ent "Number of Vertical Tic Marks",V
32: if (int(V))V)>25;dsp "Must Be Less Than 26";wait 1500;;jmp -1
33: ent "Number of Y Axis Decimal Places",W;int(W)}W
34: if W>4;dsp "Value Must Be Less Than 5";wait 1500;;jmp -1
35: ent "Enter Y Axis Label (70 Char Max)",V$
36: if len(V$))70;dsp "Label Too Long";wait 1500;;jmp -1
37: V$)M$;70)Z;gsb "CENTER"
38: M$)V$;ent "Current Year       YY",Y$;if len(Y$)#2;;jmp 0
39: ent "Clock Name      (10 Char Max)",C$
40: if len(C$))10;dsp "Name Too Long";wait 1500;;jmp -1
41: C$)M$;10)Z;gsb "CENTER"
42: M$)C$;ent "Associated Data File Name",F$
43: if len(F$))6;dsp "Name Too Long";wait 1500;;jmp -1
44: F$)M$;6)Z;gsb "CENTER"
45: M$)F$;ent "Enter Your Name  (17 Char Max)",M$
46: if len(M$))17;dsp "Name Too Long";wait 1500;;jmp -1
47: M$)M$;17)Z;gsb "CENTER"
48: M$)N$;fxd 0;for L=1 to 1000
49: 4(Lmod10=0)+(Lmod10=1)+2(Lmod10=2)+3(Lmod10=3)+4(Lmod10)3)}I
50: if Lmod100)9 and Lmod100<14;4}I
51: dsp "Enter"&str(L)&S$[I]&" X";ent "",G[L,1];if flg13;gto +7
52: if G[L,1]<H[1];dsp "Below Limit";wait 1500;;jmp -1
53: if G[L,1]>H[2];dsp "Above Limit";wait 1500;;jmp -2
54: dsp "Enter"&str(L)&S$[I]&" Y";ent "",G[L,2]
55: if G[L,2]<V[1];dsp "Below Limit";wait 1500;;jmp -1
56: if G[L,2]>V[2];dsp "Above Limit";wait 1500;;jmp -2
57: next L
58: L-1)M;for L=1 to M-1;for M=L+1 to M
59: if G[L,1]<G[M,1];;jmp 3
60: G[M,1])A;G[L,1])G[M,1];A)G[L,1]
61: G[M,2])A;G[L,2])G[M,2];A)G[L,2]
62: next M;next L
63: wrt 9,"R";red 9,T$;T$[1,2]&"-"&T$[4,5]&"-"&Y$)T$
64:
65: "INITIALIZE PLOT AREA";
66: fxd 0;hdcpy 1;pse 705;scl -7.125,0,0,10.5;plt 0,0,1
67: plt -.75,0,2;plt -.75,10.4;plt 0,10.4;plt 0,0;plt -5/16,0;plt -5/16,10.4
68: for L=1.5 to 9 by 1.5;plt -3/4,L,1;plt 0,L,2;next L;csiz 1.5,2,1,90
69: plt -7.5/16,.67,1;lbl "Date";plt -1/16,.5,1;lbl T$
70: plt -7.5/16,2,1;lbl "Program";plt -1/16,1.55,1;lbl " Gen Purpose Plot"
71: plt -7.5/16,3.35,1;lbl "Data Source";plt -5.5/16,3.35,1;lbl ""
72: plt -1/16,3.4,1;lbl "User Entry"
73: plt -9/16,5.05,1;lbl "Points";plt -5.5/16,5,1;lbl "Plotted"
74: plt -1/16,5.1,1;lbl str(M)
75: plt -7.5/16,6.4,1;lbl "  Clock  ";plt -1/16,6.4,1;lbl C$
76: plt -9/16,7.9,1;lbl "Associated ";plt -5.5/16,7.95,1;lbl "Data File"
77: plt -1/16,8.08,1;lbl F$
78: plt -7.5/16,9.25,1;lbl "Executed by"
79: plt -1/16,9.05,1;lbl M$;plt -1.375,10.4,1
80: plt -7.125,10.4,2;plt -7.125,1;plt -1.375,1;plt -1.375,10.4;flt 9
81: scl V[2],1.23913V[1]-.23913V[2],H[1]-(H[2]-H[1])/9.4,H[2]+(H[2]-H[1])/94
82:
83: "LABEL AXES";
84: for L=V[1]+((V[2]-V[1])/V)C) to V[2]-C by C;plt L,H[2],1
85: plt L,H[2]-(H[2]-H[1])/100,2;next L
86: for L=H[2]-((H[2]-H[1])/H)D) to H[1]+D by -D;plt V[2],L,1
```

```
87: plt V[2]-(V[2]-V[1])/70,L,2;next L
88: csiz 1.2,1.90;for L=V[2] to V[1] by -C
89: plt L-(V[2]-V[1])/95,H[1]-(H[2]-H[1])/17,1
90: if abs(L))=100;fxd W;lbl str(L);gto *3
91: if abs(L))=10;fxd W;lbl " "&str(L);gto *2
92: fxd W;lbl "  "&str(L)
93: if L#V[1] and L#V[2];plt L,H[1],1;plt L,H[1]+(H[2]-H[1])/100,2
94: next L;csiz 1.5,2,1,180;plt V[1]+(V[2]-V[1])/50,H[1]-(H[2]-H[1])/12,1
95: lbl V0;csiz 1.2,1,90;for L=H[1] to H[2] by D;if L=H[1] or L=H[2];;mp 2
96: plt V[1]+(V[2]-V[1])/70,L,1;plt V[1],L,2
97: plt V[1]-(V[2]-V[1])/25,L-(H[2]-H[1])/100,1;fxd B;lbl str(L);next L
98: csiz 1.5,2,1,90;plt V[1]-(V[2]-V[1])/12.25,H[1]+(H[2]-H[1])/4.71,1;lbl H0
99:
100: "PLOT DATA";
101: plt G[1,2],G[1,1],1;cplt -.33,-.15;lbl "o";plt G[1,2],G[1,1],1
102: for L=2 to N;plt G[L,2],G[L,1],2;next L;pen
103: cplt -.33,-.15;lbl "o";wtb "PRINTER",12,13,27,69
104: end
105:
106: "TIME";
107: 0)Q;for M=1 to val(T$[1,2])-1
108: if M=1 or M=3 or M=5 or M=7 or M=8 or M=10 or M=12;Q+2678400)Q;;mp 3
109: if M=4 or M=6 or M=9 or M=11;Q+2592000)Q;;mp 2
110: if M=2;Q+2419200)Q
111: next M
112: Q+86400val(T$[3,4])+3600val(T$[5,6]))Q
113: ret Q+60val(T$[7,8])+val(T$[9,10])
114:
115: "CENTER";
116: for L=1 to int((Z-len(M$))/2)
117: " "&M$)M$;next L
118: ret
#11463
```

9.33 Program: MAGPRG

```
0: "PROGRAM: ANSI STANDARD MAGNETIC TAPE PROGRAM COPY";
1:
2: "*************************************************************";
3: "*  Kevin Miller                 *  File: MAGPRG                *";
4: "*  June 22, 1981                *  Update: July 30, 1983       *";
5: "*  Naval Research Laboratory    *  Output Device: 4600         *";
6: "*  Space Applications Branch    *  Select Code: 716, 717       *";
7: "*************************************************************";
8:
9: "DECLARE DEVICES, VARIABLES, AND FORMAT STATEMENTS";
10: dsp "Insert Softcopy Graphics Tape";stp
11: idb 1;dev "MAGCTRL",716,"MAGRW",717
12: dim C,G,I,J,L,M,N,P,Q,V,F$[6],G$[6],M$[17]
13: fmt 1,z,"VOL1PROG",29x,"DIB1111007007",29x,"1"
14: fmt 2,z,c4,c17,"PROGMS00010001000100 00000 00000 000000DECFILE11A",10x
15: fmt 3,z,c4,"F0051200512",21x,"M",13x,"00",28x
16:
17: if bit(3,rds("MAGCTRL"))=1;jmp 2
18: dsp "Mount Magnetic Tape Reel";stp ;jmp -1
19: if bit(2,rds("MAGCTRL"))#1;jmp 2
20: dsp "Tape is Write Protected";stp ;jmp -3
21:
22: "GET PROGRAM AND COPY":
23: ent "Enter Program to Copy",F$;if flg13;gto "END"
24: assn F$,1,0,0;if Q#2;dsp F$&" does not exist";wait 1500;jmp -1
25: wrt "MAGCTRL","CC(0) BS(80)";F$)G$
26: for L=1 to len(F$);cap(F$[L,L]))F$[L,L];next L
27: for L=1 to len(F$);if (num(F$[L,L]))Q))>48;if Q<90;next L;jmp 2
28: "0")F$[L,L];next L
29: F$&".PGT")M$;for L=len(M$)+1 to 17;M$&" ")M$;next L
30: wrt "MAGRW.1";wrt "MAGRW.2","HDR1",M$;wrt "MAGRW.3","HDR2"
31: wrt "MAGCTRL","WE BS(512)"
32: chain G$,100,33
33: list #717,100
34: fmt 2,z,c4,c17,"PROGMS00010001000100 00000 00000 000000DECFILE11A",10x
35: fmt 3,z,c4,"F0051200512",21x,"M",13x,"00",28x
36: wrt "MAGCTRL","WE BS(80)";wrt "MAGRW.2","EOF1",M$;wrt "MAGRW.3","EOF2"
37: wrt "MAGCTRL","WE";gto "GET PROGRAM AND COPY"
38:
39: "END":
40: wrt "MAGCTRL","RW"
41: dsp "Program Copy Complete"
42: end
43:
44:
45:
46:
47:
48:
49:
50:
51:
52:
53:
54:
55:
56:
57:
58:
59:
60:
61:
62:
63:
64:
65:
66:
67:
68:
69:
70:
71:
72:
73:
74:
75:
76:
77:
78:
79:
80:
81:
82:
83:
84:
85:
86:
```

```
87:
88:
89:
90:
91:
92:
93:
94:
95:
96:
97:
98:
99:
#3463
```

9.39 Program: MAGDAT

```
0: "PROGRAM: ANSI STANDARD MAGNETIC TAPE DATA COPY"
1:
2: "*************************************************"
3: "*  Kevin Miller              *  File: MAGDAT      *"
4: "*  June 15, 1981             *  Update: July 30, 1983 *"
5: "*  Naval Research Laboratory *  Output Device: 4608  *"
6: "*  Space Applications Branch *  Select Code: 716, 717 *"
7: "*************************************************"
8:
9: "DECLARE DEVICES, VARIABLES, AND FORMAT STATEMENTS"
10: dev "MAGCTRL",716,"MAGRW",717,"CLOCK",9
11: dim C,G,I,J,L,M,N,P,Q,V,C[25],C$[25,10],D$[6],F$[6],J$[25],M$[17],T$[12]
12: fmt 1,z,"VOL1POTF",29z,"DIB1111007007",29z,"1"
13: fmt 2,z,c4,c17,"POTF  0001000100100 00000 00000 000000ECFILE11A",10z
14: fmt 3,z,c4,"F0051200512",21z,"M",13z,"00",28z
15: fmt 4,f8.3,f6.0,z,f6.0,c9,e13.4,f5.0
16: fmt 5,f4.0,c13,z,c3
17: fmt 6,z,c11,e15.6
18: fmt 7,z,f10.4
19: fmt 8,/
20: fmt 9,/,f8.4,z
21:
22: "MESSAGE"
23: spc 3;prt "----------------"," MAGNETIC TAPE","    TRANSFER"
24: prt "----------------";spc 2;prt "This program","will copy one or"
25: prt "more data files","to tape."," ","Enter CONTINUE"
26: prt "to the prompt","(Enter Data File","to Copy) to"
27: prt "write EOF's and","rewind tape.";spc 2
28: if bit(3,rds("MAGCTRL"))=1;jmp 2
29: dsp "Mount Magnetic Tape Reel";stp ;jmp -1
30: if bit(2,rds("MAGCTRL"))#1;jmp 2
31: dsp "Tape is Write Protected";stp ;jmp -3
32:
33: "GET FILE AND COPY"
34: ent "Enter Data File to Copy",F$;if flg13;gto "END"
35: asgn F$,1,1,0;if Q=1;dsp F$&" does not exist";wait 1500;jmp -1
36: asgn F$,1,1;wrt "MAGCTRL","CC(0) BS(80)"
37: for L=1 to len(F$);cap(F$[L,L]))F$[L,L];next L
38: for L=1 to len(F$);if (num(F$[L,L]))Q)>48;if Q<90;next L;jmp 2
39: "0")F$[L,L];next L
40: F$&".DAT")M$;for L=len(M$)+1 to 17;M$&" ")M$;next L
41: wrt "MAGRW.1";wrt "MAGRW.2","HDR1",M$;wrt "MAGRW.3","HDR2"
42: wrt "MAGCTRL","WE BS(512)"
43: rread 1,1,G,N,I,D$,P,J;wrt "MAGRW.4",G,N,I,D$,P,J;wrt "MAGRW.8"
44: for L=1 to J;sread 1,C[L],C$[L],J$[L,L];wrt "MAGRW.5",C[L],C$[L],J$[L,L]
45: next L;wrt "MAGRW.8";for L=1 to N;rread 1,L*4,T$,C;wrt "MAGRW.6",T$,C
46: if J=0;jmp 3
47: for M=1 to J;sread 1,V;if Mmod8=1;wrt "MAGRW.9",V;next M;jmp 2
48: wrt "MAGRW.7",V;next M
49: wrt "MAGRW.8";next L
50: wrt "MAGCTRL","WE BS(80)";wrt "MAGRW.2","EOF1",M$;wrt "MAGRW.3","EOF2"
51: wrt "MAGCTRL","WE";gto "GET FILE AND COPY"
52:
53: "END"
54: wrt "MAGCTRL","RW"
55: dsp "Tape Copy Complete"
56: end
#6333
```

10.0 Operating the Vacuum Chamber

Follow the steps below to evacuate the chamber manually:

1) press the POWER ON button
2) if the alarm goes off, press ALARM RESET
3) set the operation mode to MANUAL
4) close all valves
5) start the ROUGH and BACKER pumps
6) check the chamber doors for obvious leaks
7) open the ROUGH and FORELINE valves
8) when foreline pressure drops below 100 millitorr, turn the DIFFUSION PUMP on
9) set the red trip-point needle on the foreline pressure gauge to 100 millitorr
10) turn duplex power on
11) when chamber pressure drops below 50 millitorr, close the ROUGH valve
12) open the HIGH VAC valve
13) pressure should drop on the chamber pressure gauge
14) when it is no longer readable, turn the IONIZATION GAUGE on
15) turn the filament on whenever a pressure reading is desired, but do leave it on long as the filament has a short half life
16) set the THERMOCOUPLE select switch to the chamber base plate
17) turn THERMAL CONTROL on to control temperature inside the chamber

To turn the vacuum chamber off:

1) turn the IONIZATION GAUGE off
2) turn THERMAL CONTROL off
3) wait 1 minute for compressor fan to cut off if refrigeration was on
4) close all valves
5) turn the DIFFUSION PUMP off
6) if the chamber temperature is above 0 degrees C, the VENT VALVE can be opened
7) press the POWER OFF button

END

FILMED

3-84

DTIC